LIKELIHOOD RATIO AWARE DATA SELECTION FOR DIRECT PREFERENCE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Direct Preference Optimization (DPO) is a widely used offline preference alignment method for large language models (LLMs), and its performance is highly dependent on the quality of the preference data. Conventional selection methods prioritize the separability between paired responses, but we argue that this overlooks the inherent value of individual samples related to the true data distribution. To address this, we propose a likelihood ratio aware data selection framework that identifies high value samples by integrating distributional reliability with data separability. Our method enhances training stability by filtering out low likelihood ratio regions whose gradients are large and directionally incoherent, while achieving superior alignment performance even with a small subset of data that surpasses full dataset training. Extensive experiments across multiple large language models demonstrate that our approach consistently outperforms standard DPO, its variants, and existing selection baselines, and generalizes well to other preference learning losses.

1 INTRODUCTION

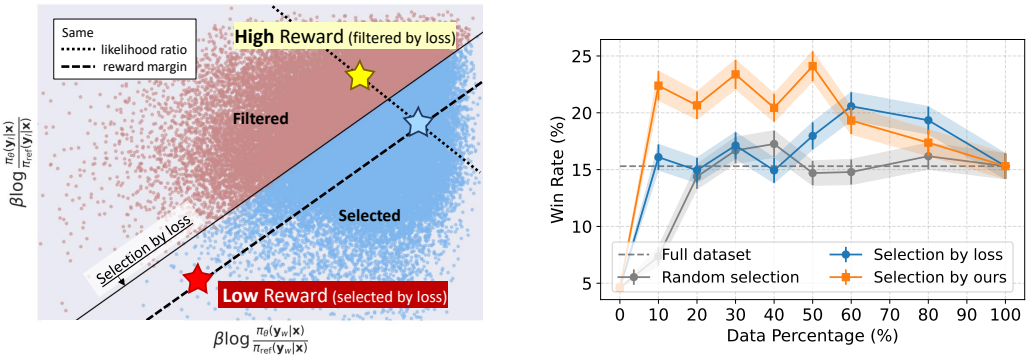
Recently, Large Language Models (LLMs) have become pervasive across a wide range of applications, from conversational assistants (Achiam et al., 2023; Team et al., 2023; Grattafiori et al., 2024) and code generation (Li et al., 2023) to complex reasoning and decision-making systems (Wei et al., 2022; Yao et al., 2023; Bubeck et al., 2023). Aligning these models with human feedback has emerged as a critical fine tuning paradigm to better capture human preferences. Among various strategies, Direct Preference Optimization (DPO) (Rafailov et al., 2023) has been proposed as a simpler and stable alternative to traditional Reinforcement Learning from Human Feedback (RLHF) (Bai et al., 2022; Ouyang et al., 2022), as it avoids specifying an explicit reward model, which may induce model bias; while maintaining comparable performance.

Similar to the importance of the reliability of the explicit reward model in the traditional RLHF, the quality of the offline preference data is important to the efficacy of DPO. However, in the real world, preference datasets are often filled with uninformative or contradictory examples (Mitchell, 2023; Gao et al., 2024; Chowdhury et al., 2024; Liang et al., 2024). While existing studies (Gao et al., 2025; Razin et al., 2024; Kong et al., 2024) have attempted to mitigate this by filtering data based on the separability between preferred and unpreferred responses, we argue that these strategies may be incomplete as it overlooks the absolute reward quality of individual responses. As illustrated in Figure 1a, multiple pairs can lie on the same reward margin line (a line with a slope of 1). While these pairs share an identical margin, they can differ significantly in their response quality. This observation raises a question on how to value a data instance with large margin and low likelihood.

To address this, we first propose that preference learning should be aligned with the original goal of finding a policy that maximizes reward, which had been shadowed by the likelihood ratio optimization in DPO. We suggest data selection criteria should consider the statistical consistency of the data generating distribution. Furthermore, we show that, under the optimal policy, the log likelihood ratio serves as a lower bound for the true reward, which motivates its use as a selection signal for estimating the inherent value of preference data. By integrating this signal with traditional separability measures, our method prioritizes preference pairs that are discriminative as well as high quality.

The practical efficiency of this approach is shown in Figure 1b. Across various data selection ratios, our method consistently identifies high value samples that provide a more reliable alignment signal than margin based filtering. Notably, a small selected subset (10–20%) already matches or surpasses

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Illustration of sample selection concept.

(b) Win rate across different selection ratios.

Figure 1: **Effectiveness of Likelihood Ratio Aware Selection.** (a) Loss based selection (red/blue) may incorrectly include low reward samples (red star) or discard high reward samples (yellow star) based exclusively on the reward margin. As illustrated, samples with identical reward margins can vary significantly in their likelihood ratios and response quality. (b) Win Rate (%) across different data selection ratios. Our likelihood ratio aware selection (Selection by ours) consistently identifies high value samples. Remarkably, a small selected subset outperforms the model trained on the full dataset (100%, dashed line).

the model trained on the full (100%) dataset, demonstrating that filtering uninformative pairs is more critical for alignment than increasing total data volume.

Our contributions are as follows:

1) We show that, under the optimal policy, the likelihood ratio serves as a principled lower bound for the true reward, providing a mathematical basis for estimating data value.

2) We show that low likelihood ratio samples produce gradients that are large in norm and directionally incoherent, and that filtering them reduces gradient variance.

3) Our method improves alignment performance across various large language models empirically. Notably, training on a small selected subset (10–20%) already surpasses the full dataset (100%), confirming the effectiveness of our selection criteria.

2 Preliminary

2.1 RLHF and DPO

We consider an auto-regressive language model π_θ as a policy model that generates a response $\mathbf{y} = [y^1, \ldots, y^T]$ given a prompt $\mathbf{x}$. The probability of generating $\mathbf{y}$ given $\mathbf{x}$ is defined as $\pi_\theta(\mathbf{y}|\mathbf{x}) = \prod_{k=1}^{T} \pi_\theta(y^k|\mathbf{x}, \mathbf{y}^{<k})$, where T denotes the token length of a response $\mathbf{y}$.

RLHF is the standard framework for aligning large language models with human preferences. It typically consists of three stages: (1) Supervised Fine-Tuning (SFT) to adapt a pretrained language model to reference model π_{ref}, (2) Reward Model (RM) r_ϕ training, whose preference training data tuple form is as $(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)$, where $\mathbf{x}$ is a prompt, $\mathbf{y}_w$ is a preferred response and $\mathbf{y}_l$ is an unpreferred response, and (3) Reinforcement Learning (RL) fine-tuning, where the policy model is optimized to maximize the reward, often using algorithms such as PPO (Schulman et al., 2017; Christiano et al., 2017).

The optimization objective of training π_θ is given as follows:

$$\max_{\pi_\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}\sim\pi_\theta(\mathbf{y}|\mathbf{x})}[r_{\phi^*}(\mathbf{x},\mathbf{y})] - \beta \text{KL}(\pi_\theta(\mathbf{y}|\mathbf{x})||\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})) \tag{1}$$

and the optimal policy π_{θ^*} of Eq. 1 can be theoretically expressed in the following closed form:

$$\pi_{\theta^*}(\mathbf{y}|\mathbf{x}) = \frac{\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})\exp\left(\frac{1}{\beta} r_{\phi^*}(\mathbf{x},\mathbf{y})\right)}{Z(\mathbf{x})}, \text{ where } Z(\mathbf{x}) = \sum_{\mathbf{y}} \pi_{\text{ref}}(\mathbf{y}|\mathbf{x})\exp\left(\frac{1}{\beta} r_{\phi^*}(\mathbf{x},\mathbf{y})\right) \tag{2}$$

In practice, as the ground truth reward r_{ϕ^*} is unknown and $Z(\mathbf{x})$ is intractable, RLHF cannot directly implement Eq. 2. Consequently, it necessitates an iterative training pipeline, which is often sensitive to reward model quality and prone to instable (Gao et al., 2023; Rafailov et al., 2023). To address these limitations, Direct Preference Optimization (DPO) (Rafailov et al., 2023) reformulates alignment fine-tuning as a supervised learning that directly optimizes π_θ using preference data not relying on an explicit reward model. The reward model of Eq. 2 is expressed in closed form as:

$$r_{\phi^*}(\mathbf{x}, \mathbf{y}) = \beta \log \frac{\pi_{\theta^*}(\mathbf{y}|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})} + \beta \log Z(\mathbf{x}) \tag{3}$$

By substituting the optimal policy π_{θ^*} in Eq. 3 with the current policy π_θ, DPO reparameterizes the reward function as $r_\theta(\mathbf{x}, \mathbf{y}) = \beta \log \frac{\pi_\theta(\mathbf{y}|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})} + \beta \log Z(\mathbf{x})$. Integrating it into the Bradley-Terry model (Bradley & Terry, 1952), the empirical loss of DPO with regard to the dataset D is given as:

$$\mathcal{L}_{\text{DPO}}(D; \theta) := -\frac{1}{N} \sum_{i=1}^{N} \log \sigma \left(\beta \log \frac{\pi_\theta(\mathbf{y}_w^i|\mathbf{x}^i)}{\pi_{\text{ref}}(\mathbf{y}_w^i|\mathbf{x}^i)} - \beta \log \frac{\pi_\theta(\mathbf{y}_l^i|\mathbf{x}^i)}{\pi_{\text{ref}}(\mathbf{y}_l^i|\mathbf{x}^i)} \right). \tag{4}$$

$D := \{(\mathbf{x}^i, \mathbf{y}_w^i, \mathbf{y}_l^i)\}_{i=1}^N$ denotes a preference dataset which is assumed to be sampled from the underlying oracle distribution by the following process: $\mathbf{x} \sim p^*(\mathbf{x})$, $(\mathbf{y}_1, \mathbf{y}_2) \sim p^*(\mathbf{y} \mid \mathbf{x})$ and $(\mathbf{y}_w, \mathbf{y}_l)$ are obtained by ordering $(\mathbf{y}_1, \mathbf{y}_2)$ according to $p^*(\mathbf{y}_1 \succ \mathbf{y}_2 \mid \mathbf{x}, \mathbf{y}_1, \mathbf{y}_2)$. This formulation has achieved comparable alignment performance with improved stability and efficiency.

2.2 Preference Data Selection

Although DPO simplifies alignment, its performance heavily depends on the quality of the preference dataset. Since all preference pairs are treated equally in the objective, noted as an unweighted summation over N in Eq. 4; low-quality samples can mislead the optimization to wrong alignment.

To address this issue, prior work proposes preference data filtering based on discrepancies between paired responses, using criteria such as hidden embedding geometry (Razin et al., 2024), perplexity differences (Kong et al., 2024), or validation loss (Gao et al., 2025). For example, Gao et al. (2025) use DPO loss as a proxy for sample difficulty and discard samples with large losses. Since large DPO loss is equivalent to the small reward margin,[1] this strategy prioritizes samples with large reward margins. It means the criterion can retain samples when both responses yield low rewards as long as they are well separated. While it can be useful for preference learning, such samples may provide limited signal for improving absolute reward.

3 Motivation for Likelihood Ratio Aware Data Selection

3.1 Statistical Consistency

While DPO optimizes preferences as Eq. 4, the RLHF objective is to learn a policy that maximizes the expected reward as Eq. 1. This implies that for DPO to converge to the same optimal policy of RLHF, the preference signal should be reliable at least for the responses generated from π^*. Given that preference data are in practice limited, the essential samples are those that are more likely to be generated from π^*. Considering that the preference data generation process typically involves (1) prompt sampling, (2) response generation, and (3) preference labeling, it means the preference signal should be assigned a higher weight if the response pair $(\mathbf{y}_1, \mathbf{y}_2)$ is generated with high probability.

Importance sampling (Robert et al., 1999; Katharopoulos & Fleuret, 2018) is a well established weighting scheme that uses the likelihood ratio to account for discrepancies between different distributions. We apply this principle to preference data selection by utilizing the likelihood ratio. Directly reweighting the objective by this ratio turns out to be numerically unstable (Section 5.3), so we instead use it as a selection signal. This motivates a detailed analysis of the likelihood ratio's behavior within our dataset.

Figure 2 presents the correlation matrix within the preference training dataset, between: (1) total log likelihood ratio (**LLR**): $\log \frac{\pi_\theta(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})} + \log \frac{\pi_\theta(\mathbf{y}_l|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l|\mathbf{x})}$,[2] (2) DPO loss (**Loss**):

[1] Remark 4.1

[2] By the independence of response pairs, the joint likelihood is defined as $p(\mathbf{y}_w, \mathbf{y}_l|\mathbf{x}) = p(\mathbf{y}_w|\mathbf{x})p(\mathbf{y}_l|\mathbf{x})$

$-\log\sigma\left(\beta\log\frac{\pi_\theta(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})}-\beta\log\frac{\pi_\theta(\mathbf{y}_l|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l|\mathbf{x})}\right)$, and (3) negative log likelihood (**NLL**): computed as the length normalized sum $-\frac{1}{|y_w|}\log\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})-\frac{1}{|y_l|}\log\pi_{\text{ref}}(\mathbf{y}_l|\mathbf{x})$.[3] It shows that the correlation between (1) LLR and Loss, (2) LLR and NLL is low. This indicates that selecting preference data solely based on Loss ignores whether those samples are likely under the true data distribution. The low correlation between LLR and NLL further underscores the complexity of the alignment process. If LLR were strongly correlated with NLL, the model would merely be reinforcing high probability regions of the reference policy. However, our result shows that data selection depending on π_{ref} may be problematic fundamentally due to the distributional discrepancy between π_{ref} and π_θ.

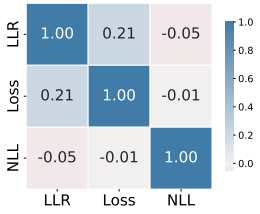


Figure 2: Correlation matrix of log likelihood ratio (LLR), DPO loss (Loss) and negative log likelihood (NLL).

3.2 Relation to Reward

Beyond distributional alignment, ensuring the selected samples reflect reward related properties is essential for stable policy optimization. Figure 1a illustrates an example when the training samples are filtered based on the DPO loss. As shown in the figure, even when two different samples exhibit the same reward margin, their absolute reward value can differ significantly. This indicates that such samples contribute differently to policy optimization despite having identical preference signals.

We underline that the likelihood ratio serves as a lower bound of the reward, as formally stated in Theorem 3.1. We denote $\frac{\pi_\theta(\mathbf{y}|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})}$ as $\text{lr}_\theta(\mathbf{y})$ for convenience. (See Appendix A.1 for proof.)

Theorem 3.1. *For any response* $\mathbf{y}$*, the centered reward under the optimal policy is lower bounded by the log likelihood ratio* $\log\text{lr}_{\theta^*}(\mathbf{y})$.

$$r_{\phi^*}(\mathbf{x},\mathbf{y})-\mathbb{E}_{\mathbf{y}\sim\pi_{\text{ref}}}[r_{\phi^*}(\mathbf{x},\mathbf{y})]\geq\beta\log\text{lr}_{\theta^*}(\mathbf{y})$$

Remark 3.2 (Practical instantiation). Theorem 3.1 is stated for the optimal policy π_{θ^*}, which is unknown in practice. Our implementation (Algorithm 1) replaces it with a proxy $\pi_{\bar{\theta}}$ obtained by cross-fitted DPO training on disjoint data splits. This proxy is not arbitrary: $\beta\log\text{lr}_{\bar{\theta}}(\mathbf{y})$ is exactly the implicit reward that DPO itself assigns to $\mathbf{y}$ (Eq. 3).

We do not claim that the proxy attains π_{θ^*}; the theorem serves as the motivation for the criterion rather than a bound on the proxy-based score. The dependence on the proxy is instead examined empirically: the selection quality is insensitive to the number of proxy models m (Table 4) and to the choice of proxy architecture (Fig. 16).

This lower bound motivates prioritizing high likelihood ratio samples: whereas filtering by the DPO loss alone is agnostic to the absolute quality of individual responses, adding the likelihood ratio drives policy optimization toward high-reward, high-quality data, complementing the distributional adjustment of the previous section.

3.3 Learning Dynamics from Gradient Analyses

While Figure 2 shows that the pearson correlation between the DPO loss and the likelihood ratio is negligible, the two are coupled during optimization. Differentiating the DPO loss with respect to the likelihood ratio of each response gives the following (proof in Appendix A.2).

Proposition 3.3. *For a preference pair* $(\mathbf{x},\mathbf{y}_w,\mathbf{y}_l)\in D$*, the magnitude of the DPO loss derivative with respect to the likelihood ratio (*lr_θ*) is inversely proportional to* lr_θ.

$$\left|\frac{d\mathcal{L}_{\text{DPO}}(\mathbf{x},\mathbf{y}_w,\mathbf{y}_l;\theta)}{d\text{lr}_\theta(\mathbf{y})}\right|\propto\frac{1}{\text{lr}_\theta(\mathbf{y})}$$

where $\mathbf{y}\in\{\mathbf{y}_w,\mathbf{y}_l\}$.

[3]Following the previous researches (Rafailov et al., 2023; Liu et al., 2024), we utilize π_{ref} and π_θ as proxies for data distribution and the true distribution, respectively.

As detailed in Appendix A.2, the derivative also carries the factor $(1 - p_\theta(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})) \in (0, 1)$, but this is bounded, whereas $1/\mathrm{lr}_\theta \in (0, +\infty)$ grows without bound as the likelihood ratio approaches zero: the loss is most sensitive exactly in the low likelihood ratio region. In parameter space this factor is absorbed, since $\nabla_\theta \mathrm{lr}_\theta = \mathrm{lr}_\theta \nabla_\theta \log \mathrm{lr}_\theta$ yields the standard form $\nabla_\theta \mathcal{L}_{\mathrm{DPO}} = -\beta(1 - p_\theta)[\nabla_\theta \log \pi_\theta(\mathbf{y}_w|\mathbf{x}) - \nabla_\theta \log \pi_\theta(\mathbf{y}_l|\mathbf{x})]$, so we measure the actual updates directly.

Low likelihood ratio samples destabilize training through two properties of their gradients. First, they are large: across all three architectures, the average gradient norm of the *Small* lr_θ group exceeds that of the *Large* group (Figure 3). Second, they are incoherent: per-sample DPO gradients of the n lowest-lr_θ samples are mutually near-orthogonal (cosine ≈ 0), while those of the n highest-lr_θ samples are consistently more aligned at every n (Table 1). Large updates pointing in inconsistent directions add variance to the batch gradient rather than a coherent signal. Filtering out this region is therefore not only a data quality decision but a variance reduction step for DPO training.

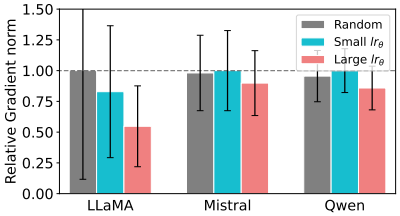


Figure 3: Relative gradient norm of the *Small* and *Large* lr_θ groups (each 50% of the training set), scaled so that the maximum in each model is 1.

Table 1: Mean pairwise cosine similarity of per-sample DPO gradients (LLaMA-3-8B, last four layers) within the n lowest- and n highest-lr_θ samples.

n	Low lr_θ	High lr_θ
10	−0.002	0.073
20	0.001	0.061
30	0.001	0.052

4 Likelihood Ratio Aware Preference Data Selection

Building on the analysis regarding the likelihood ratio, we now introduce our preference data selection strategy. For convenience, we denote the reward margin as

$$\mathrm{RM}_\theta(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) := \beta \log \frac{\pi_\theta(\mathbf{y}_w|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}_w|\mathbf{x})} - \beta \log \frac{\pi_\theta(\mathbf{y}_l|\mathbf{x})}{\pi_{\mathrm{ref}}(\mathbf{y}_l|\mathbf{x})} = \beta \log \mathrm{lr}_\theta(\mathbf{y}_w) - \beta \log \mathrm{lr}_\theta(\mathbf{y}_l) \quad (5)$$

Data selection based on the DPO loss can be reformulated in terms of reward margin, as Remark 4.1.

Remark 4.1. Data selection with small DPO loss is equivalent to selecting samples with large reward margin. Formally,

$$\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta) \le \tau \Leftrightarrow \mathrm{RM}_\theta(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \ge -\log(e^\tau - 1),$$

as $\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta) = -\log \sigma\left(\mathrm{RM}_\theta(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)\right)$. τ is a selection threshold.

We incorporate the likelihood ratio into the selection criterion by combining it with the reward margin through linear weighting.

Definition 4.2. Let λ_1 and λ_2 be weighting parameters to the log likelihood ratios of the winning and losing responses, respectively. Define the scoring function $s_{\theta,(\lambda_1,\lambda_2)}$ of a sample $(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)$ as:

$$s_{\theta,(\lambda_1,\lambda_2)}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) := \mathrm{RM}_\theta(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) + \lambda_1 \beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \lambda_2 \beta \log \mathrm{lr}_\theta(\mathbf{y}_l)$$

Tuning (λ_1, λ_2) introduces a redundant degree of freedom: the selected subset depends only on the ordering of scores, and the following proposition shows that a single parameter λ reproduces every such ordering (proof in Appendix A.3).

Proposition 4.3. *Let $D(s; k) \subseteq D$ be the subset of top-k samples selected according to the score s. Then, the selection criterion (λ_1, λ_2) is reducible to a single parameter λ, if $\lambda_1 \ge -1$. Formally,*

$$\forall k,\ \exists \lambda \in \mathbb{R} \quad \textit{s.t.} \quad D(s_{\theta,(\lambda_1,\lambda_2)}; k) = D(s_{\theta,(\lambda,\lambda)}; k).$$

Writing $s^\lambda_\theta := s_{\theta,(\lambda,\lambda)}$, expanding the reward margin gives

$$s^\lambda_\theta(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) = (\lambda + 1)\, \beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + (\lambda - 1)\, \beta \log \mathrm{lr}_\theta(\mathbf{y}_l), \quad (6)$$

so λ directly sets how much each response's own likelihood ratio counts: $\lambda = 0$ recovers margin-only selection, $\lambda \to \infty$ recovers selection by the summed likelihood ratio, and for $0 < \lambda < 1$ the winning response is rewarded for being likely while the losing response is still penalized for it. Nonlinear combinations are examined in Appendix C.4.

Definition 4.4. Given a trade-off parameter λ and a selection threshold τ, we define the *likelihood ratio aware selected* preference dataset $S_\theta^{\lambda,\tau}$ as:

$$S_\theta^{\lambda,\tau} = \{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \in D \mid s_\theta^\lambda(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \geq \tau\}$$

Following the previous research (Gao et al., 2025), we calculated π_θ by averaging predictions from multiple (m) models to reduce the estimation error. Algorithm 1 summarizes the overall process for sample selection.

Algorithm 1: Likelihood Ratio Aware Data Selection

Input: π_{ref} (reference model), D (dataset), m (the number of models for π_θ estimation), τ (sample selection rate), λ (trade off parameter)

Output: $S_\theta^{\lambda,\tau}$ (Selected preference dataset)

for $t = 1, ..., m$ **do**

  Randomly split D into two disjoint subsets D_t^1 and D_t^2

  Let $\theta_t^1 := \arg\min_\theta \mathcal{L}_{\text{DPO}}(D_t^1; \theta)$ and $\theta_t^2 := \arg\min_\theta \mathcal{L}_{\text{DPO}}(D_t^2; \theta)$

  Calculate the likelihood ratios as $\text{lr}_{\theta_t}(\mathbf{y}) := \begin{cases} \text{lr}_{\theta_t^2}(\mathbf{y}) & \text{if } (\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \in D_t^1 \\ \text{lr}_{\theta_t^1}(\mathbf{y}) & \text{if } (\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \in D_t^2 \end{cases}$

end

Compute $\text{lr}_{\bar{\theta}}(\mathbf{y}^i) = \frac{1}{m} \sum_{t=1}^m \text{lr}_{\theta_t}(\mathbf{y}^i)$ for all $i \in \{1, \ldots, |D|\}$ and $\mathbf{y}^i \in \{\mathbf{y}_w^i, \mathbf{y}_l^i\}$.

Construct selected subset $S_{\bar{\theta}}^{\lambda,\tau}$ according to definition 4.4 using $\bar{\theta}$.

5 Experiment

To empirically validate the effectiveness of our likelihood ratio aware data selection framework, we conduct experiments and report the performances of the trained models selected with our selection in this section.

5.1 Settings

We directly utilize high-quality SFT checkpoints which are already available from the community. Specifically, we employ LLaMA-3-8B-SFT (Meng et al., 2024)[4], Mistral-7B-SFT (Tunstall et al., 2023)[5], and Qwen-2.5-7B-SFT (Hu et al., 2024)[6] as our reference model. For preference data, we mainly use the UltraFeedback-Binarized dataset (Cui et al., 2024), which is a common benchmark in alignment research (Gao et al., 2025; Wu et al., 2025; Meng et al., 2024). Unless otherwise specified, experimental results for our sample selection strategy are based on a 50% sampling rate following Gao et al. (2025) and $\lambda = 2/3$ (selected by grid search on LLaMA-3 and fixed across all models). Detailed configurations and hyperparameters are in Appendix C.2.

5.2 Results

As summarized in Table 2, our likelihood ratio aware selection strategy achieves superior alignment performance in the vast majority of cases and remain highly competitive, including AlpacaEval 2 (Dubois et al., 2024), Arena-Hard (Li et al., 2025), and MT-Bench (Zheng et al., 2023).

We first apply our sample selection strategy to the standard DPO. While numerous DPO variants have been introduced, our results (Table 2) indicate that standard DPO often maintains more stable performance across diverse tasks. By integrating our data-centric refinement with this robust foundation, we demonstrate how strategic data selection can significantly elevate the performance of a strong, widely-adopted baseline without requiring complex changes to the objective function. Next, we extend our selection strategy to DPO variants as Table 3. It shows our selection strategy improves learning with different objectives also. For the baseline explanations, please refer to Appendix B.

[4]https://huggingface.co/princeton-nlp/Llama-3-Base-8B-SFT

[5]https://huggingface.co/HuggingFaceH4/mistral-7b-sft-beta

[6]https://huggingface.co/glorgao/Qwen2.5-7B-SFT

Table 2: AlpacaEval 2 (Dubois et al., 2024), Arena-Hard (Li et al., 2025) and MT-Bench (Zheng et al., 2023). For LLaMA and Mistral, performances of *General DPO variations* are from SimPO (Meng et al., 2024). As SimPO does not report them for Qwen-2.5, those cells are missing. Other rows are rerun within our pipeline. Random means random selection of training dataset. **Bold** means the best performances and underline means second best of each column. LC and WR means Length-Controlled winning rate and Winning Rate, respectively.

Method	LLaMA-3-8B-SFT				Mistral-7B-SFT				Qwen-2.5-7B-SFT			
	AlpacaEval2		Arena	MT-Bench	AlpacaEval2		Arena	MT-Bench	AlpacaEval2		Arena	MT-Bench
	LC(%)	WR(%)	WR(%)	Score	LC(%)	WR(%)	WR(%)	Score	LC(%)	WR(%)	WR(%)	Score
SFT	6.2	4.6	3.3	6.6	8.4	6.2	1.3	6.3	10.4	5.2	7.2	8.0
General DPO variations												
RRHF (Yuan et al., 2023)	12.1	10.1	6.3	7.0	11.6	10.2	5.8	6.7	–	–	–	–
SLiC-HF (Zhao et al., 2023)	12.3	13.7	6.0	7.6	10.9	8.9	7.3	7.4	–	–	–	–
DPO (Rafailov et al., 2023)	18.2	15.5	15.9	7.7	15.1	12.5	10.4	7.3	18.6	17.4	30.5	8.9
IPO (Azar et al., 2024)	14.4	14.2	17.8	7.4	11.8	9.4	7.5	7.2	10.3	10.9	22.8	7.8
CPO (Xu et al., 2024)	10.8	8.1	5.8	7.4	9.8	8.9	6.9	6.8	–	–	–	–
KTO (Ethayarajh et al., 2024)	14.2	12.4	12.5	7.8	13.1	9.1	5.6	7.0	–	–	–	–
ORPO (Hong et al., 2024)	12.2	10.6	10.8	7.6	14.7	12.2	7.0	7.3	–	–	–	–
R-DPO (Park et al., 2024)	17.6	14.4	17.2	7.5	17.4	12.8	8.0	7.4	–	–	–	–
SimPO (Meng et al., 2024)	22.0	20.3	23.4	7.7	21.5	20.8	16.6	7.3	17.1	20.6	29.2	8.5
Soft preference learning												
CDPO (Mitchell, 2023)	17.7	14.8	25.2	7.6	15.2	12.7	8.5	7.3	18.6	16.0	25.8	8.8
RDPO (Chowdhury et al., 2024)	17.1	11.7	26.7	7.7	16.9	16.7	9.4	7.5	19.9	18.5	31.8	8.9
GDPO (Furuta et al., 2024)	7.9	4.8	11.3	6.5	7.4	4.8	2.0	5.6	10.6	5.6	7.5	8.0
Data selection												
Random (50%)	16.7	14.7	32.5	7.6	16.6	14.2	12.5	7.7	18.1	16.2	26.6	8.9
CHES (50%) (Razin et al., 2024)	9.4	13.4	18.1	7.3	13.8	16.1	7.6	7.2	14.1	16.5	22.4	8.6
PPL* (50%) (Gao et al., 2025)	19.9	18.0	31.2	7.4	18.2	15.8	9.7	7.6	18.0	16.7	31.2	**9.2**
Loss* (50%) (Gao et al., 2025)	21.1	18.0	29.2	7.5	20.9	21.1	14.3	7.5	25.5	**24.7**	35.3	8.6
TIF (64%) (Zhang et al., 2026)	26.6	22.4	34.8	7.6	24.1	18.2	17.9	7.6	–	–	–	–
Ours (10%)	19.2	24.1	30.4	7.8	23.8	22.0	**21.0**	7.5	12.8	8.6	–	–
Ours (50%)	**27.7**	**28.8**	**39.5**	**8.0**	**24.2**	**23.6**	15.2	**7.8**	**25.9**	24.2	**38.5**	9.0

The results demonstrate that our method consistently achieves superior alignment performance, effectively transferring the intuitions of using the likelihood ratio for preference data selection into tangible empirical gains. Please refer to Appendix C.4 for more results.

Selection Ratio Figure 1b illustrates the win rates of models across various training data selection ratios.[7] When the model is trained with the data selected by our criterion, it consistently outperforms models trained on data selected by baselines (loss based (Base) and random (Random)). This performance gain is particularly evident at lower selection ratios. Remarkably, on LLaMA-3 and Mistral only 10% of the data selected by our criterion already surpasses the full (100%) dataset model; on Qwen-2.5, whose likelihood ratio distribution is narrower, 20% suffices (Appendix 9). This suggests that our criterion successfully identifies high quality samples while filtering out potentially noisy or redundant data that can degrade model performances. As the selection ratio increases beyond 60%, the gap to the baselines narrows, as expected, since the high quality samples are eventually included in the baseline sets. That our method reaches its peak earlier (10-50%) demonstrates its ability to identify core samples.

Sensitivity analysis on m Gao et al. (2025) suggested that the performances of the resulting model are improved with the increased number of models until $m = 3$ for π_θ estimation for their selection criterion. Following them, we mainly experimented based on $m = 3$ settings. However, $m = 1$ is even enough to get high quality samples for our likelihood ratio aware selection as in Table 4. It shows the robustness of likelihood ratio aware selection to the estimated model and underlines the effectiveness of likelihood ratio awareness for preference data selection.

Performances with λ change To understand how the reliance on the likelihood ratio during data selection affects alignment performance, we analyzed the performances with changing the trade-off parameter λ in our selection metric as Figure 4 (**RM+LLR**). We also present the **LLR** curve to show the independent impact of the likelihood ratio as a selection criterion, isolating its contribution from our full strategy. LLR at $\lambda = 0$ means random selection, while RM+LLR at $\lambda = 0$ corresponds to loss-based selection (Base). We also include the performance of a model trained on the full (100%) dataset as a reference (**Full dataset**).

[7]For fair comparison, we fixed $m = 3$ for this figure.

Table 3: LLR-aware selection application to DPO variants.

Method	LC(%)	WR(%)
SimPO	21.50	20.80
+Ours	**22.57 (+1.07)**	**25.82 (+5.02)**
CDPO	17.70	14.80
+Ours	**24.65 (+6.95)**	**24.67 (+9.87)**
RDPO	17.11	11.72
+Ours	**23.54 (+6.43)**	**20.44 (+8.72)**

Table 4: Sensitivity of m for π_θ estimation.

Selection (%)	m	LC(%)	WR(%)
10	1	18.13	22.99
	2	**19.23**	**24.05**
	3	16.12	22.56
50	1	**27.69**	**28.84**
	2	27.22	25.08
	3	25.10	24.10

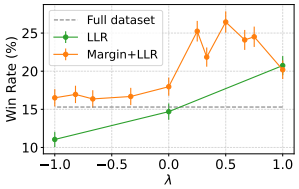


Figure 4: Win Rate with varying λ. (SFT baseline: 4.6%)

Another question in our investigation is whether the model benefits more from samples with high likelihood ratios or low ones. In related fields such as coreset selection (Coleman et al., 2019; Paul et al., 2021), active learning (Yoo & Kweon, 2019; Ducoffe & Precioso, 2018), and online batch sampling (Mindermann et al., 2022; Lin et al., 2024), samples who represent patterns the model has not yet learned are argued to be informative. Under this logic, one might expect that prioritizing low likelihood ratio samples (negative λ) would yield better results. However, Figure 4 shows that performance tends to improve as λ increases for both LLR and RM+LLR curves. This indicates that samples with high likelihood ratios provide a more reliable signal in this task.

The performance of RM+LLR peaks at $0 < \lambda < 1$, and Eq. 6 explains why. For $\lambda < 1$ the coefficient $(\lambda - 1)$ on $\log \mathrm{lr}_\theta(\mathbf{y}_l)$ is negative, so the score still penalizes losing responses that the policy finds likely, preserving the preference signal. At $\lambda \geq 1$ this coefficient becomes non-negative: the score no longer distinguishes a pair by which response lost, and selection degenerates toward likelihood alone, which explains the degradation beyond the peak. The flatter response of Qwen-2.5 (Fig. 9c) again reflects its narrower likelihood ratio distribution (Appendix Table 10). Full results for all models are in Appendix C.4.

5.3 Analyses on Likelihood Ratio Aware Selection

Correlation and Estimation Reliability Figure 5 is the correlation matrix among the log likelihood ratios of $\mathbf{y}_l$, standard deviations, and the aggregate log likelihood ratio (LLR) of each preference pair. Please refer to Appendix C.4 for $\mathbf{y}_w$ as Figure 10.

We observe a significant negative correlation (-0.97) between the log likelihood ratios and their corresponding standard deviations. This relation is further substantiated by the hexbin plots in Figure 11 (in Appendix C.4), where the data points closely align with the $y = -x$ trajectory. It confirms that the negative correlation is not a systematic artifact but reflects an intrinsic property of the samples. Crucially, the standard deviation decreases as LLR increases. These findings indicate that samples with higher likelihood ratios are associated with lower estimation uncertainty, meaning our sample selection criterion relies on more robust signals.

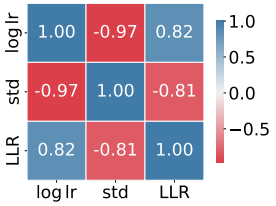


Figure 5: Correlation matrices between the log likelihood ratio (log lr), the standard deviation (std) and total log likelihood ratio (LLR).

Estimated Likelihood Ratio Given that the likelihood ratio between the reference and current policies can serve as an importance sampling weight, a natural extension would be to incorporate these ratios directly into the DPO training objective. However, our empirical investigation reveals that this weighting approach leads to training instability. As shown in Figure 12 (in Appendix C.4), the training accuracy stagnates near 50%, indicating a failure to learn. We further experimented by applying these weights exclusively to a subset of samples pre-selected based on small losses, yet this configuration also resulted in training failure.

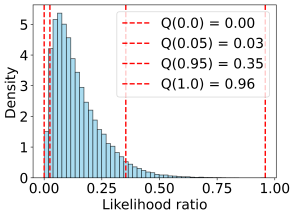


Figure 6: Histogram of likelihood ratio. Vertical dashed lines represent quantiles.

The histogram of likelihood ratios in Figure 6 provides a plausible explanation for these failures. The distribution is highly skewed, with 95% of the weights concentrated below 0.35. Such a severe imbalance suggests that naively applying these weights significantly reduces the effective sample size, as a small number of samples with disproportionately

Table 5: Group-wise statistics with the sample percentages (%), average reward margins and log likelihood ratios, and the maximum perplexity of the winning and losing responses of each group (Perplexity $\times 10^3$).

	Selected by		Sample (%)	Reward margin	LLR	Perplexity	
	Base	Ours				$\mathbf{y}_w$	$\mathbf{y}_l$
[1]	✓	✓	37.16	**0.86**	-1.07	3.00	31.68
[2]	✓	✗	12.84	0.59	**-1.52**	**4.13**	**38.66**
[3]	✗	✓	12.84	0.10	-0.60	1.04	9.07
[4]	✗	✗	37.16	-0.24	-1.19	5.26	19.08

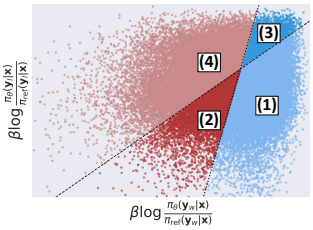


Figure 7: Visualization of selected and filtered samples (loss based selection (Base) vs. Ours).

large weights may dominate the gradient, while the majority of samples become negligible. This numerical instability likely hinders the optimization process. These findings justify our choice of a selection based approach. For more explanations, please refer to Appendix C.4.

Statistics of Selected Samples Table 5 provides a comparative analysis of the statistical properties of the samples selected by either our strategy or loss with Figure 7 as scatter plots to illustrate where selected and filtered samples are located. Specifically, we show the percentage of samples in each group, the average reward margin of each group, the average total log likelihood ratios of responses in each group (LLR), the worst (maximum) perplexity of winning and losing responses in each group. While the overlap between the two methods is 37.16%, the samples selected exclusively by one method correspond to 34.55% $(12.84/37.16)$ relative to the intersection.

The results align with our core intuition: samples selected exclusively by our method exhibit smaller reward margins, whereas those selected solely by the loss based criterion show significantly lower log likelihood ratios. Notably, our approach substantially improves the worst case perplexity, reducing it from approximately $4,130$ to $1,040$ for winning responses group and from $38,660$ to $9,070$ for the losing responses. These findings suggest that our strategy effectively filters out corrupted or out-of-distribution samples that simple margin based metrics might overlook.

5.4 Computation Complexity

As we showed in Figure 1b and Table 2, training on 100% data leads to suboptimal compared to carefully selected 10%. This underlines the importance of sample selection for better model alignment. Here, we further compare the required computation time.

Table 6: Computation time comparison: DPO vs. Ours total pipeline. Step 1 is π_θ estimation, Step 2 is sample selection, and Step 3 is policy training with selected samples.

Process		Selection (%)	Time (s)	WR (%)
Original DPO		100	27,077.42	15.3
Ours	Step 1	10	3,710.92	**23.0**
	Step 2		19.09	
	Step 3		2,710.13	
	Total		**6,440.15**	

Table 6 shows the significant reduction in actual training data (to 10%) effectively compensates for step 1 and step 2 cost, while its performance is superior to the origianl DPO training with full dataset. Our total pipeline is approximately $4.2\times$ faster than full dataset DPO training (6,440s vs. 27,077s).

6 Conclusion

In this paper, we proposed a new data selection criterion for direct preference learning, moving beyond loss based metrics to a strategy that accounts for likelihood ratios. Our analysis confirms that likelihood ratios serve as a more robust indicator of sample quality than loss alone, as they are linked to the true data distribution. By prioritizing samples with high likelihood ratios, we established a selection criterion for the preference-alignment objective that reduces gradient variance and improves alignment performance. Our empirical findings demonstrate that this likelihood ratio aware sample selection approach maintains its effectiveness across different model architectures. Ultimately, our work suggests that considering the likelihood perspective provides a more principled foundation for data curation, offering an efficient path for aligning large language models with human preferences.

AI USE STATEMENT

Among the subtasks whose disclosure is required, we did not use generative AI tools for any: generating synthetic data, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting the proof writing, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, interpreting results were either carried out by the authors or not related to this paper.

Among the subtasks whose disclosure is recommended, we used generative AI tools to edit the paper for readability, including sentences and paragraphs revising for clarity and grammar, to suggest a structure for parts of the paper to a limited extent, and to look up minor coding details when producing figures (e.g., matplotlib commands for adjusting font sizes). We did not use generative AI tools for the other recommended subtasks: summarizing or identifying literature, formatting references, or brainstorming and discovering research topics. Our implementation builds on publicly available open-source code and related work was identified through a manual literature survey.

We have reviewed all AI-assisted work. Every AI-edited sentence was checked by the authors against the underlying results and revised when needed, and the structural suggestions were largely not adopted, with the final organization decided by the authors. No claim, number, proof, or artifact in this paper originates from a generative AI tool. We take responsibility for the final content of this work, including text produced with the aid of generative AI.

ETHICS STATEMENT

This work studies data selection for preference-based alignment of large language models. It does not involve human subjects, personally identifiable information, or the collection or release of new data; all experiments use publicly available models and preference datasets under their respective licenses. Our method selects a subset of existing preference pairs and does not introduce new sources of bias beyond those already present in the underlying datasets, though, as with any alignment method, the behavior of the resulting models depends on the quality and biases of that preference data. We foresee no additional ethical concerns, conflicts of interest, or risks of harmful application specific to this work beyond the general considerations that apply to alignment research on large language models.

REPRODUCIBILITY STATEMENT

All experiments use publicly available SFT checkpoints (LLaMA-3-8B, Mistral-7B, Qwen-2.5-7B) and the publicly available UltraFeedback-Binarized dataset, with additional experiments on the Argilla and code-preference datasets; sources for all models and datasets are cited in Experiment Settings. The full selection procedure is given as Algorithm 1, and the trade-off parameter λ and number of proxy models m, together with our tuning protocol, are explained in the experiment section. Complete training and evaluation configurations, including all hyperparameters and decoding settings for AlpacaEval 2, Arena-Hard, and MT-Bench, are reported in Appendix C.2 and Appendix C.3. All theoretical claims (Theorem 3.1, Propositions 3.3 and 4.3) are stated with their assumptions and proved in full in Appendix A.1–A.3. We will release our code upon publication.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Argilla. Ultrafeedback binarized preferences (cleaned). `https://huggingface.co/datasets/argilla/ultrafeedback-binarized-preferences-cleaned`, 2024. Hugging Face dataset.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learn-

ing from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pp. 4447–4455. PMLR, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Sébastien Bubeck et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably robust dpo: Aligning language models with noisy feedback. In *International Conference on Machine Learning*, pp. 42258–42274. PMLR, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. In *International Conference on Learning Representations*, 2019.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=BOorDpKHiJ.

Xun Deng, Han Zhong, Rui Ai, Fuli Feng, Zheng Wang, and Xiangnan He. Less is more: Improving llm alignment via preference data selection. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Yann Dubois, Percy Liang, and Tatsunori Hashimoto. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024.

Melanie Ducoffe and Frederic Precioso. Adversarial active learning for deep networks: a margin based approach. *arXiv preprint arXiv:1802.09841*, 2018.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Hiroki Furuta, Kuang-Huei Lee, Shixiang Shane Gu, Yutaka Matsuo, Aleksandra Faust, Heiga Zen, and Izzeddin Gur. Geometric-averaged preference optimization for soft preference labels. *Advances in Neural Information Processing Systems*, 37:57076–57114, 2024.

Chengqian Gao, Haonan Li, Liu Liu, Zeke Xie, Peilin Zhao, et al. Principled data selection for alignment: The hidden risks of difficult examples. In *Forty-second International Conference on Machine Learning*, 2025.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pp. 10835–10866. PMLR, 2023.

Yang Gao, Dana Alon, and Donald Metzler. Impact of preference noise on the alignment performance of generative language models. *arXiv preprint arXiv:2404.09824*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2024.

Yulan Hu, Qingyang Li, Sheng Ouyang, Ge Chen, Kaihui Chen, Lijun Mei, Xucheng Ye, Fuzheng Zhang, and Yong Liu. Towards comprehensive preference data collection for reward modeling. *arXiv preprint arXiv:2406.16486*, 2024.

Kexin Huang, Junkang Wu, Ziqian Chen, Xue Wang, Jinyang Gao, Bolin Ding, Jiancan Wu, Xiangnan He, and Xiang Wang. Larger or smaller reward margins to select preferences for llm alignment? In *Forty-second International Conference on Machine Learning*, 2025.

Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International conference on machine learning*, pp. 2525–2534. PMLR, 2018.

Keyi Kong, Xilie Xu, Di Wang, Jingfeng Zhang, and Mohan S Kankanhalli. Perplexity-aware correction for robust alignment with noisy preferences. *Advances in Neural Information Processing Systems*, 37:28296–28321, 2024.

Raymond Li et al. Starcoder: may the source be with you! *Transactions on Machine Learning Research*, 2023.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34209–34231. PMLR, 13–19 Jul 2025.

Xize Liang, Chao Chen, Jie Wang, Yue Wu, Zhihang Fu, Zhihao Shi, Feng Wu, and Jieping Ye. Robust preference optimization with provable noise tolerance for llms. *CoRR*, 2024.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, Weizhu Chen, et al. Not all tokens are what you need for pretraining. *Advances in Neural Information Processing Systems*, 37:29029–29063, 2024.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572, 2023.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pp. 6448–6458. PMLR, 2020.

Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235, 2024.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pp. 15630–15649. PMLR, 2022.

Eric Mitchell. A note on dpo with noisy preferences and relationship to ipo. `https://ericmitchell.ai/cdpo.pdf`, 2023. Technical note.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling length from quality in direct preference optimization. *arXiv preprint arXiv:2403.19159*, 2024.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1944–1952, 2017.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34:20596–20607, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Noam Razin, Sadhika Malladi, Adithya Bhaskar, Danqi Chen, Sanjeev Arora, and Boris Hanin. Unintentional unalignment: Likelihood displacement in direct preference optimization. In *The Thirteenth International Conference on Learning Representations*, 2024.

Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment, 2023. *URL https://arxiv. org/abs/2310.16944*, 6, 2023.

Jason Wei et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 2022.

Yue Wu, Zhiqing Sun, Rina Hughes, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. In *International Conference on Learning Representations*, volume 2025, pp. 91558–91582, 2025.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*, 2024.

Shunyu Yao et al. React: Synergizing reasoning and acting in language models. *ICLR*, 2023.

Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 93–102, 2019.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems*, 36:10935–10950, 2023.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Zizhuo Zhang, Qizhou Wang, Shanshan Ye, Jianing Zhu, Jiangchao Yao, Bo Han, and Masashi Sugiyama. Towards understanding valuable preference data for large language model alignment. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=FUp0KeEEBs.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

A PROOF

A.1 THEOREM 3.1

Theorem A.1. *Let π_{θ^*} denote the optimal policy and r_{ϕ^*} be the reward function defined in Eq. 3. Then, for any response $\mathbf{y}$, the centered reward is lower bounded by the log likelihood ratio* $\log \mathrm{lr}_{\theta^*}(\mathbf{y})$.

$$r_{\phi^*}(\mathbf{x}, \mathbf{y}) - \mathbb{E}_{\mathbf{y}\sim\pi_{\mathrm{ref}}}[r_{\phi^*}(\mathbf{x}, \mathbf{y})] \geq \beta \log \mathrm{lr}_{\theta^*}(\mathbf{y})$$

Proof. From Eq. 3,

$$\begin{aligned} r_{\phi^*}(\mathbf{x}, \mathbf{y}) &= \beta \log \mathrm{lr}_{\theta^*}(\mathbf{y}) + \beta \log Z(\mathbf{x}) && (7)\\ &= \beta \log \mathrm{lr}_{\theta^*}(\mathbf{y}) + \beta \log \sum_{\mathbf{y}} \pi_{\mathrm{ref}}(\mathbf{y}|\mathbf{x}) \exp\left(\frac{1}{\beta} r_{\phi^*}(\mathbf{x}, \mathbf{y})\right) && (8)\\ &\geq \beta \log \mathrm{lr}_{\theta^*}(\mathbf{y}) + \mathbb{E}_{\mathbf{y}\sim\pi_{\mathrm{ref}}}[r_{\phi^*}(\mathbf{x}, \mathbf{y})] && (9) \end{aligned}$$

by Jensen's inequality. It means that if the prompt $\mathbf{x}$ is determined, the likelihood ratio provides a lower bound on the reward as $\mathbb{E}_{\pi_{\mathrm{ref}}}[r^*(\mathbf{x}, \mathbf{y})]$ becomes a constant. □

A.2 PROPOSITION 3.3

Proposition A.2. *For a preference pair $(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \in D$, the magnitude of the DPO loss derivative with respect to the likelihood ratio (lr_θ) is inversely proportional to lr_θ.*

$$\left| \frac{d\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta)}{d\mathrm{lr}_\theta(\mathbf{y})} \right| \propto \frac{1}{\mathrm{lr}_\theta(\mathbf{y})}$$

where $\mathbf{y} \in \{\mathbf{y}_w, \mathbf{y}_l\}$.

Proof. The preference probability is $p_\theta(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x}) = \sigma\left(\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) - \beta \log \mathrm{lr}_\theta(\mathbf{y}_l)\right)$ and the DPO loss $\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta)$ becomes

$$\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta) = -\log \sigma\left(\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) - \beta \log \mathrm{lr}_\theta(\mathbf{y}_l)\right) = \log\left(1 + \exp(-\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \beta \log \mathrm{lr}_\theta(\mathbf{y}_l))\right) \tag{10}$$

Then,

$$\begin{aligned} \frac{d\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta)}{d\mathrm{lr}_\theta(\mathbf{y}_w)} &= \frac{\left(\exp(-\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \beta \log \mathrm{lr}_\theta(\mathbf{y}_l))\right)'}{1 + \exp(-\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \beta \log \mathrm{lr}_\theta(\mathbf{y}_l))} && (11)\\ &= \frac{-\frac{\beta}{\mathrm{lr}_\theta(\mathbf{y}_w)} \cdot \left(\exp(-\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \beta \log \mathrm{lr}_\theta(\mathbf{y}_l))\right)}{1 + \exp(-\beta \log \mathrm{lr}_\theta(\mathbf{y}_w) + \beta \log \mathrm{lr}_\theta(\mathbf{y}_l))} && (12)\\ &= -\frac{\beta}{\mathrm{lr}_\theta(\mathbf{y}_w)}\left(1 - p_\theta(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})\right) && (13) \end{aligned}$$

Similarly,

$$\frac{d\mathcal{L}_{\mathrm{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \theta)}{d\mathrm{lr}_\theta(\mathbf{y}_l)} = \frac{\beta}{\mathrm{lr}_\theta(\mathbf{y}_l)}\left(1 - p_\theta(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})\right) \tag{14}$$

□

A.3 PROPOSITION 4.3

Proposition A.3. *Let $D(s;k) \subseteq D$ be the subset of top-k samples selected according to the score s. Then, the selection criterion (λ_1, λ_2) is reducible to a single parameter λ, if $\lambda_1 \geq -1$. Formally,*

$$\forall k,\ \exists \lambda \in \mathbb{R} \quad \textit{s.t.} \quad D(s_{\theta,(\lambda_1,\lambda_2)};k) = D(s_{\theta,(\lambda,\lambda)};k).$$

Proof. The sample selection criterion $s_{\theta,(\lambda_1,\lambda_2)}(\mathbf{x},\mathbf{y}_w,\mathbf{y}_l) = \mathrm{RM}_\theta(\mathbf{x},\mathbf{y}_w,\mathbf{y}_l) + \lambda_1 \log \mathrm{lr}_\theta(\mathbf{y}_w) + \lambda_2 \log \mathrm{lr}_\theta(\mathbf{y}_l)$ can be written as:

$$(1+\lambda_1)\cdot \log \mathrm{lr}_\theta(\mathbf{y}_w) + (-1+\lambda_2)\cdot \log \mathrm{lr}_\theta(\mathbf{y}_l) \tag{15}$$

Define a single parameter λ to jointly weight $\log \mathrm{lr}_\theta(\mathbf{y}_w)$ and $\log \mathrm{lr}_\theta(\mathbf{y}_l)$ as follows:

$$\mathrm{RM}_\theta(\mathbf{x},\mathbf{y}_w,\mathbf{y}_l) + \lambda\left(\log \mathrm{lr}_\theta(\mathbf{y}_w) + \log \mathrm{lr}_\theta(\mathbf{y}_l)\right) = (1+\lambda)\cdot \log \mathrm{lr}_\theta(\mathbf{y}_w) + (-1+\lambda)\cdot \log \mathrm{lr}_\theta(\mathbf{y}_l) \tag{16}$$

For the two selection criteria of Eq. 15 and Eq. 16 to be equivalent:

$$\frac{1+\lambda}{1+\lambda_1} = \frac{-1+\lambda}{-1+\lambda_2} \Leftrightarrow \lambda = \frac{\lambda_1+\lambda_2}{2+\lambda_1-\lambda_2} \tag{17}$$

To preserve the direction of the inequality, we restrict $\lambda, \lambda_1 \geq -1$.

(1) For $\lambda_1 > -1$ and $\lambda_2 \neq 1$, Eq. 17 holds.

(2) For $\lambda_1 = -1$, it will work if $\lambda_2 < 1$ and $\lambda = -1$.

(3) For $\lambda_2 = 1$, it will work if $\lambda_1 > -1$ and $\lambda = 1$.

(4) For the case when $(1+\lambda_1) = (-1+\lambda_2)$, it will be equivalent if $\lambda \to \infty$. □

B RELATED WORKS

B.1 ALIGNMENT OF LARGE LANGUAGE MODELS

Recent advances in large pre-trained language models have been accompanied by significant efforts to align these models with human preferences. Reinforcement Learning from Human Feedback (RLHF) has emerged as a standard paradigm for LLM alignment (Stiennon et al., 2020; Ouyang et al., 2022). In RLHF, a reward model is trained on human preference data and subsequently used to guide KL-regularized reinforcement learning fine-tuning with algorithms such as PPO (Schulman et al., 2017) or GRPO (Shao et al., 2024). Despite its effectiveness, RLHF introduces substantial computational and memory overhead because it requires propagating additional components, such as reward and value models, during fine-tuning.

Several approaches, such as RRHF (Yuan et al., 2023) and DPO (Rafailov et al., 2023), have been proposed to mitigate these limitations by leveraging implicit reward formulations instead of explicit reward modeling (Yuan et al., 2023; Rafailov et al., 2023; Meng et al., 2024; Ethayarajh et al., 2024). These methods typically incorporate human preference data directly into policy optimization. Specifically, IPO (Azar et al., 2024) addresses the overfitting tendencies of DPO by replacing the log-sigmoid loss of DPO with a quadratic loss; CPO (Xu et al., 2024) utilizes a contrastive loss to widen the probability gap between preferred and rejected responses, specifically tailored for tasks like machine translation; KTO (Ethayarajh et al., 2024) optimizes the model based on the Kahneman-Tversky prospect theory, allowing for alignment using individual good or bad labels rather than explicit preference pairs; ORPO (Hong et al., 2024) merges supervised fine-tuning and preference alignment into a single step by incorporating an odds ratio-based penalty directly into the loss function; R-DPO (Park et al., 2024) addresses the length bias in DPO by disentangling response length from quality, ensuring the model doesn't simply favor longer responses as a proxy for better

ones; SimPO (Meng et al., 2024) is a reference-free approach that uses the average log-probability of tokens as a reward implicit, enforcing a target margin between preferred and rejected outputs.

Eliminating the need for explicit reward models greatly simplifies the training process and reduces resource consumption. However, since preference data directly influences the policy optimization objective, the performance of these approaches is sensitive to the quality of the preference datasets (Chowdhury et al., 2024).

B.2 Noisy Preference Learning

As the quality of the preference dataset is crucial for the efficacy of RLHF and DPO, the presence of noise and ambiguity remains a significant challenge. This line of research is also related in the long-standing problem of label noise classification (Patrini et al., 2017; Lukasik et al., 2020; Zhang et al., 2021), where the goal is to learn a robust classifier despite corrupted training labels. Inspired by these classical spirits, several studies have attempted to adapt robust optimization techniques to the preference learning setting. CDPO (Mitchell, 2023) introduces a label smoothing technique into the DPO loss for the binary preference one-hot label $(1, 0)$ as $(\epsilon, 1-\epsilon)$ to prevent the model from over-adjusting to potentially noisy binary labels. RDPO (Chowdhury et al., 2024) assumes that preference labels are flipped with a certain probability ϵ and assume a preference label transition matrix. Then they employ a backward loss correction method to recover the loss to be consistent to that with true preference signal. It is a technique well-established in the transition matrix literature of noisy label classification. GDPO (Furuta et al., 2024) utilizes assumes the winner and loser outputs are sampled from the weighted geometric averaging of LLM policies, and the weighted geometric averaging can be considered as a regularization to push the large likelihood down to small when the soft preference is far from 1. Therefore, they can address potential overfitting to noisy signals. Furthermore, ROPO (Liang et al., 2024) focuses the robust loss: instead of using the cross entropy loss for learning the preference probability, they directly maximize the preference probability with their loss function as $1 - p(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})$.

While these methods modify the objective function to tolerate noise, our approach provides a cleaner learning signal by proactively filtering out uninformative samples before the optimization begins.

B.3 Preference Data Selection

Given the sensitivity of policy optimization to data quality, selecting a high-quality subset from a large-scale preference pool has become a crucial research direction. Unlike the aforementioned robust loss functions, data selection methods aim to identify and utilize only the most informative samples for alignment.

SelectiveDPO (Gao et al., 2025) utilizes the validation loss of the DPO objective itself as a proxy for sample difficulty, advocating for the use of "easy" samples to ensure stable convergence. Another approach, CHES (Razin et al., 2024), focuses on the geometric properties of model representations; it calculates a score based on the inner product and L_2 norm of hidden embeddings to mitigate likelihood unalignment, thereby selecting samples that better represent the underlying preference structure. BeeS (Deng et al., 2025) suggests reward margin aggregation from multiple sources (e.g. implicit reward and explicit reward) for sample selection. M_{AP} (Huang et al., 2025) quantifies the gap from the model's current implicit reward margin to the target explicit reward margin to prevent assessing data quality based on either explicit or implicit reward margins. TIF (Zhang et al., 2026) considers the influence of samples, suggesting to select samples based on the loss difference and reward margin.

C Experiment

C.1 Computational Environment

All training experiments in this study were conducted on a compute node featuring 4 × A100 (80GB) GPUs. While LoRA was specifically used during the reward estimation phase (Step 1 in algorithm 1), we employed full-parameter fine-tuning for all subsequent model training on the same infras-

tructure. For those looking to replicate our 7B model benchmarks, the training typically takes about 8 hours for Llama-3 and Qwen, and 6 hours for Mistral on our 4-GPU setup.

C.2 Implementation Details

SFT Following (Gao et al., 2025), we get the well trained Supervised Fine Tuning (SFT) model of LLaMA-3-8B https://huggingface.co/princeton-nlp/Llama-3-Base-8B-SFT, Mistral-7B https://huggingface.co/HuggingFaceH4/mistral-7b-sft-beta and Qwen-2.5 https://huggingface.co/glorgao/Qwen2.5-7B-SFT. For the main paper, we show the alignment results starting from base models. For the instruction models as checkpoint, we report results in Section C.4.

π_θ estimation To ensure a fair comparison and maintain consistency with prior literature, our hyperparameter configurations were primarily adapted from (Gao et al., 2025). While we kept architectural parameters constant across all models, the learning rate and batch sizes were specifically aligned with those reported in the baseline studies for Llama-3, Mistral, and Qwen-2.5 respectively as established in previous benchmarks. Detailed configurations are as Table 7.

Table 7: Hyperparameters for π_θ estimation. Configurations are adapted from prior studies to ensure consistency in benchmarking.

Hyperparameter	**Llama-3 (8B)**	**Mistral (7B)**	**Qwen-2.5 (7B)**
Base Model	Llama-3-Base-SFT	Mistral-7B-SFT-Beta	Qwen2.5-7B-SFT
Learning Rate	3.0×10^{-5}	3.0×10^{-5}	1.0×10^{-5}
Optimizer	Paged AdamW (32-bit)		
LR Scheduler	Cosine (Warmup ratio: 0.1)		
DPO β	0.01		
LoRA r / α	16 / 16		
Per-device Batch Size	2	8	4
Grad. Accumulation	64	16	16
Total Batch Size	512	512	256
Max Length	1024 (Prompt: 512)		

Training We followed (Gao et al., 2025) for the training stage also. Detailed configurations are as Table 8.

Table 8: Final Hyperparameter Configurations for DPO Training.

Hyperparameter	**Llama-3 (8B)**	**Mistral (7B)**	**Qwen-2.5 (7B)**
Base Model	Llama-3-Base-SFT	Mistral-7B-SFT-Beta	Qwen2.5-7B-SFT
Learning Rate	1.0×10^{-6}	1.0×10^{-6}	8.0×10^{-7}
DPO β	0.01		
Optimizer	Paged AdamW (32-bit)		
LR Scheduler	Cosine (Warmup ratio: 0.1)		
Per-device Batch Size	4	8	4
Grad. Accumulation	4	2	4
Total Batch Size	128		
Max Length	1024 (Prompt: 512)		
Seed	42		

C.3 Evaluation Details

To comprehensively assess the performance of our models on open-ended instruction-following tasks, we utilize three representative benchmarks: AlpacaEval 2 (Dubois et al., 2024), Arena-Hard (Li et al., 2025) and MT-Bench (Zheng et al., 2023). All reported scores were computed in strict adherence to the specific evaluation protocols established by their respective frameworks. Note that

for certain baseline models, including the LLaMA and Mistral series, we report the results directly sourced from SimPO (Meng et al., 2024) to ensure a consistent comparison with existing literature.

AlpacaEval 2 comprises 805 questions aggregated from five diverse datasets. We evaluate our models through pairwise comparisons against a reference model, GPT-4-1106-Preview. Performances are reported using both the raw Win Rate (WR) and the Length-Controlled Win Rate (LC), with the latter specifically designed to mitigate the response length bias from model verbosity. Regarding generation, we adopt sampling-based decoding strategies aligned with community standards: temperatures are set to 0.7 for Mistral and Qwen-2.5 models, and 0.9 for Llama-3.

MT-Bench is designed to test multi-turn dialogue proficiency through 80 questions spanning eight distinct categories. In this framework, GPT-4 acts as the primary judge, scoring response quality on a scale of 1 to 10. We report the average scores based on judgments from GPT-4, while adhering to the official decoding configurations where sampling temperatures vary according to the specific model architecture being evaluated.

Arena-Hard v0.1 serves as an advanced extension of MT-Bench, featuring 500 sophisticated queries focused on technical problem-solving. Following the protocol outlined by (Meng et al., 2024), we prioritize reproducibility by employing default greedy decoding across all experimental settings. The benchmark measures the Win Rate (WR) relative to GPT-4-0314, utilizing GPT-4-1106-Preview as an automated judge to ensure evaluative consistency.

C.4 Additional Experimental Results

Preference Accuracy While we argue that the objective of RLHF is to find a policy to maximize the true reward rather than narrowly optimizing preference labels, it remains essential to verify that our likelihood based sample selection does not compromise the current preference learning mechanism. Since the DPO framework is inherently built upon preference optimization, our selection strategy must prove it does not break the model's ability to learn these signals.

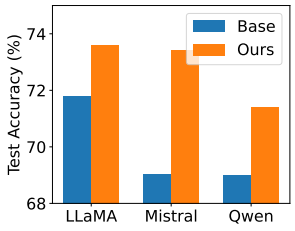
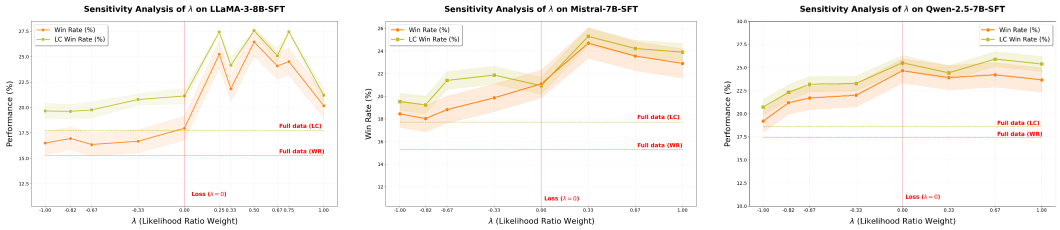


Figure 8: Preference accuracy for UltraFeedback test dataset across models.

We evaluate the preference accuracy on the UltraFeedback-Binarized test dataset as Figure 8. Across LLaMA-3, Mistral, and Qwen models, our method consistently improves test accuracy compared to the loss based baseline. This empirical evidence confirms that our focus on likelihood ratio does not compromise the preference learning of DPO. Rather, it demonstrates that our method keeps the fundamental ability of a model to distinguish preference signals.

Performances with λ change We reported the performance comparison with different λ for LLaMA-3 as figure 4. Here, we show full results including Mistral and Qwen.



(a) LLaMA-3 (b) Mistral (c) Qwen-2.5

Figure 9: Performances with λ changes. In this figure, orange and green line each means Win rate and Length Controlled (LC) Win rate, respectively. $\lambda = 0$ means samples are selected only by loss. Dotted lines for each color mean performances with full samples.

As λ is designed to balance absolute data quality (Likelihood Ratio) with relative preference (Reward Margin), the performance variations across different λ value validate again the importance of

incorporating likelihood ratio for data selection. Notably, for all three models, the performance at $\lambda = 0$ (loss only) is markedly lower than the peak performance achieved when $\lambda > 0$, highlighting that relative preference alone cannot fully capture the informative value of a sample.

The performance of RM+LLR consistently reaches its peak at $0 < \lambda < 1$ for all evaluated three models, surpassing the performance of the full dataset baseline (indicated by the dotted lines). It means samples with high likelihood ratios are more useful for training in this task.

Selection ratio on Qwen-2.5 Table 9 reports AlpacaEval 2 for Qwen-2.5 across selection ratios. Unlike LLaMA-3 and Mistral, 10% does not yet surpass full-dataset DPO; 20% does, with a clear margin at 50%. As discussed below, this reflects Qwen's narrower likelihood ratio distribution.

Table 9: AlpacaEval 2 on Qwen-2.5 across selection ratios

	LC (%)	WR (%)
SFT	10.4	5.2
Ours (10%)	12.8	8.6
Ours (20%)	19.9	18.1
Ours (50%)	**25.9**	**24.2**
DPO (100%)	18.6	17.4

Likelihood ratio distribution across models Table 10 reports percentiles of the estimated log likelihood ratio on the training set. The spread (p95−p5) for Qwen-2.5 is roughly one third of that for LLaMA-3 for both winning and losing responses. Since our criterion separates samples by their likelihood ratios, a compressed distribution leaves less room to isolate high value pairs; this accounts for Qwen's higher required selection ratio (Table 9) and its flatter response to λ (Fig. 9c).

Table 10: Percentiles of the log likelihood ratio on the training set.

Response	Model	p5	p25	p50	p75	p95	p95−p5
Winning	LLaMA-3	−1.823	−1.180	−0.826	−0.572	−0.298	1.525
	Qwen-2.5	−0.449	−0.205	−0.080	−0.005	0.138	0.587
Losing	LLaMA-3	−2.310	−1.693	−1.258	−0.859	−0.478	1.832
	Qwen-2.5	−0.566	−0.329	−0.193	−0.085	0.029	0.595

Nonlinear score combinations We do not claim that the linear score of Definition 4.4 is optimal; it is the simplest instantiation of jointly considering margin and likelihood ratio, with the per-response interpretation of Eq. 6. To probe whether a nonlinear interaction helps, we add to the (scale-normalized) linear score a multiplicative term $\gamma \cdot \mathrm{RM}_\theta \cdot \mathrm{LLR}_\theta$, which is large when the two signals share the same sign; $\gamma > 0$ adds and $\gamma < 0$ subtracts the interaction. Table 11 shows no consistent gain: the two selection ratios favor opposite signs of γ, and at 50% the small LC change is offset by a lower WR. Given that the interaction adds a hyperparameter without reliable benefit, we retain the linear form and leave nonlinear or two-stage criteria to future work.

Table 11: Effect of a multiplicative interaction term (LLaMA-3-8B, AlpacaEval 2).

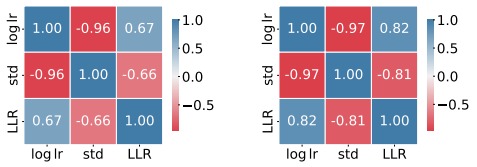
γ	10% LC	10% WR	50% LC	50% WR
-1	20.08	19.89	21.84	21.25
0 (linear)	16.12	22.56	25.10	24.10
$+1$	19.22	19.27	25.98	21.16

Variance across training seeds To verify that our gains are not artifacts of a single run, we repeat DPO training on LLaMA-3 with three seeds (41/42/43) for both selection ratios (Table 12). Every individual seed exceeds full-dataset DPO on both LC and WR, and the standard deviation is well below the margin over the baselines. Repeating full fine-tuning for every model and baseline (≈6–8h per run on 4×A100) is impractical, so we report seed statistics for LLaMA-3 only.

Visualization on each sample's LLR and std In the main paper, we reported that the correlation of log likelihood ratio (log lr) and its standard deviation is significantly negative (-0.97). In this section, we report the correlation matrix of winning response case also as Figure 10. Similar to the losing response case, the correlation between the log likelihood ratios and their corresponding

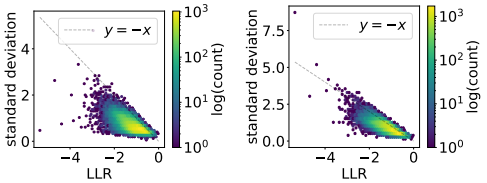
Table 12: AlpacaEval 2 mean ± std over three training seeds (LLaMA-3)

	LC (%)	WR (%)
Full data (100%)	18.2	15.5
Ours (10%)	19.70 ± 0.69	24.28 ± 0.68
Ours (50%)	**27.49 ± 0.25**	**26.42 ± 2.11**



(a) Winning response (b) Losing response

Figure 10: Correlation matrix



(a) Winning response (b) Losing response

Figure 11: Hexbin plots showing the relation between LLR and the standard deviation σ.

standard deviations is negative also for winning response, again confirming that samples with higher likelihood ratios are associated with lower estimation uncertainty.

We visualize each data points as Figure 11. Dots show a triangularly scattered shape where the $y = -x$ line acts as an empirical upper bound, further illustrating how higher LLRs strictly constrain the estimation variance.

Importance Sampling Concretely, the importance-sampling (IS) variant trains a likelihood-ratio-weighted DPO loss,

$$\mathcal{L}_{\mathrm{IS}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N} w^i \log \sigma\left(\beta \log \frac{\pi_\theta(\mathbf{y}_w^i|\mathbf{x}^i)}{\pi_{\mathrm{ref}}(\mathbf{y}_w^i|\mathbf{x}^i)} - \beta \log \frac{\pi_\theta(\mathbf{y}_l^i|\mathbf{x}^i)}{\pi_{\mathrm{ref}}(\mathbf{y}_l^i|\mathbf{x}^i)}\right),$$
$$\text{where } w^i = \mathrm{sg}\left[\frac{\pi_{\bar{\theta}}(\mathbf{y}_w^i|\mathbf{x}^i)\,\pi_{\bar{\theta}}(\mathbf{y}_l^i|\mathbf{x}^i)}{\pi_{\mathrm{ref}}(\mathbf{y}_w^i|\mathbf{x}^i)\,\pi_{\mathrm{ref}}(\mathbf{y}_l^i|\mathbf{x}^i)}\right] \tag{18}$$

$\mathrm{sg}[\cdot]$ denotes stop-gradient and $\pi_{\bar{\theta}}$ is the same proxy used for selection, so w^i is a fixed weight. The IS+selection variant applies the same weighted loss restricted to the selected subset $S_{\bar{\theta}}^{\lambda,\tau}$, with N replaced by $|S_{\bar{\theta}}^{\lambda,\tau}|$. Figure 12 is the training accuracy when applying likelihood ratio directly as an IS weight. As we can see in the figure, it resulted in training failure, with the training accuracy stagnating near 50%. We further experimented by applying IS weights exclusively to samples pre-selected based on small losses; however, this configuration also resulted in training failure. Consequently, these results justify our choice of a selection based strategy.

As an another alternative to select samples with importance weight, we conducted rejection sampling (Liu et al., 2024) and show its result as Table 13. As we can see in the table, although rejection sampling yields reasonable performance and avoids the training collapse observed with direct IS, it still falls short of our proposed strategy. This performance gap arises because our approach utilizes a more sophisticated selection criterion based on a weighted sum of reward margin and log likelihood ratio. These results further justify prioritizing samples through our method over simply reweighting or sampling.

More Statistics of Selected Samples Similar to the main paper, figure 13 and 14 are the visualization of where selected and filtered samples are located, specifically comparing the selection patterns of (1) perplexity based approach against ours, and (2) CHES based approach against ours. Tables provide their respective statistics. For visualization, the samples [1] selected by both metrics, [2] [3] selected by one metric and rejected by another metric, and [4] rejected by both metrics do not appear to be clearly clustered.

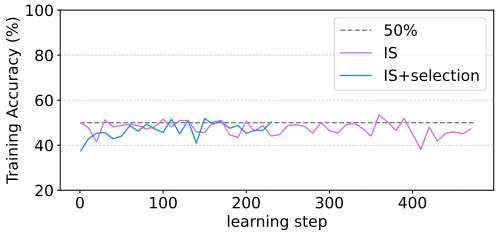


Figure 12: IS means Importance Sampling. Training accuracy oscillates around 50% during the learning process.

Table 13: Rejection Sampling (RS) Alpacaeval 2 result. Max is the case where M is set such that the acceptance probability of the sample with the highest score is normalized to 1.

Selected Data(%)	Method	LC(%)	WR(%)
–	SFT	6.2	4.6
Max	RS	12.5	11.0
10	RS	8.7	5.7
	Ours	**19.2**	**24.1**
50	RS	22.9	21.4
	Ours	**27.7**	**28.8**

Table 14: Statistics of PPL vs. Ours (Perplexity $\times 10^3$).

	Selected by		Sample (%)	Reward margin	LLR	Perplexity	
	PPL	Ours				$\mathbf{y}_w$	$\mathbf{y}_l$
[1]	✓	✓	25.85	0.60	-0.89	3.00	0.89
[2]	✓	✗	24.15	-0.09	**-1.24**	**5.26**	1.57
[3]	✗	✓	24.15	**0.72**	-1.02	0.68	31.68
[4]	✗	✗	25.85	0.03	-1.30	0.95	**38.66**

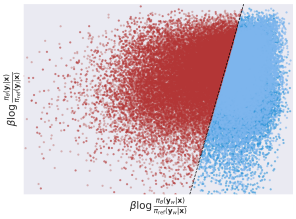


Figure 13: PPL vs. Ours

Table 15: Statistics of CHES vs. Ours (Perplexity $\times 10^3$).

	Selected by		Sample (%)	Reward margin	LLR	Perplexity	
	CHES	Ours				$\mathbf{y}_w$	$\mathbf{y}_l$
[1]	✓	✓	26.07	**0.76**	-0.98	0.04	31.68
[2]	✓	✗	23.93	0.02	**-1.35**	0.36	**38.66**
[3]	✗	✓	23.93	0.56	-0.92	**3.00**	12.65
[4]	✗	✗	26.07	-0.07	-1.21	5.26	7.44

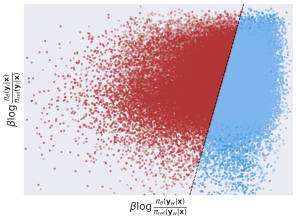
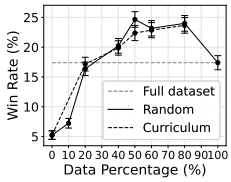


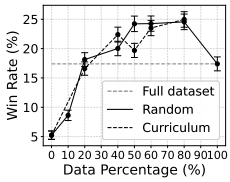
Figure 14: CHES vs. Ours

Different from the statistics of Table 5, the samples are almost equally distributed across the four groups, with each comprising approximately 25% of the total. The reward margins are inconsistent for Table 14; the reward margins of pairs of selected samples by PPL are sometimes negative, meaning the reward of winning responses is smaller than that of losing responses. Notably, group [2] exhibits a negative reward margin (-0.09), proving that PPL fails to filter flipped preferences.

Although Table 15 does not show such noisy preferences in the selected subset, the maximum perplexity for the losing response is large (38.66×10^3), meaning possible failure of filtering low quality samples.

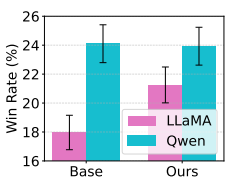


(a) Base

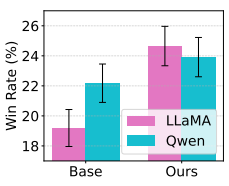


(b) Ours

Figure 15: Random vs. Curriculum learning: The shape of the line is the training order.



(a) Model: LLaMA



(b) Model: Qwen

Figure 16: Cross-model validation: Ours shows lower variance across different labelers.

Curriculum Learning Curriculum learning means presenting training samples in an increasing order of difficulty, where easy samples are introduced first, followed by harder samples. While prior studies have suggested that curriculum learning can enhance model performance by gradually increasing task difficulty (Gao et al., 2025), our empirical results present a more nuanced perspective. Figure 15 compares the win rates of models trained based on a curriculum learning strategy versus a random batch baseline across various data percentages. For the loss-based selection strategy, the curriculum presents samples with the smallest loss first and those with larger loss later. For our proposed strategy, the curriculum introduces samples with the largest reward margin and likelihood ratio first.

As illustrated in Figure 15, the curriculum strategy does not yield a significant performance gain over the random baseline. Across various data percentages, the win rates of both approaches remain statistically indistinguishable, with extensively overlapping confidence intervals. This suggests that the inherent quality of the selected samples is a more dominant factor in DPO performance than the specific chronological order of their presentation. These findings imply that a robust sample selection criterion may negate the necessity for complex curriculum scheduling.

Cross Model Generalization As π_θ estimation is essential for sample selection, it entails computational overhead. Thus, if π_θ estimations from a surrogate model can be effectively reused, it can have high practical value. Figure 16 illustrates the performance variation when samples are selected based on π_θ estimations provided by different preference labelers (LLaMA and Qwen).

In the Loss based selection baseline, we observe a substantial performance gap depending on the choice of the labeler, indicating a lack of robustness to the labeler's intrinsic bias. In contrast, our method consistently maintains high win rates with minimal variance across different labelers. This result is intuitive, as our selection criterion accounts for the intrinsic likelihood ratio of the sample itself, rather than relying solely on the model's current learning state (i.e., loss). By capturing a more universal quality of preference data that transcends specific model architectures, our approach allows for the efficient reuse of previously saved reward estimations, significantly reducing the redundant computation required for model-specific estimation.

Performances: Downstream Tasks Our criterion is designed for the preference-alignment objective on which the proxy $\pi_{\bar{\theta}}$ is trained. A natural concern is whether alignment-oriented selection damages general capabilities. Tables 16 and 17 report seven general benchmarks for LLaMA-3-8B and Mistral-7B across the full range of λ.

Three observations follow. First, likelihood ratio aware selection does not degrade general capability: for every $\lambda \in \{-1, -2/3, 2/3, 1\}$ the seven-benchmark average stays within about one point of full-data DPO on both models. Second, margin-only selection ($\lambda = 0$) *does* degrade it, most visibly on GSM8K (48.75 on LLaMA-3; 14.48 on Mistral), and adding the likelihood ratio term with either sign restores performance to the full-data level. Third, there is no systematic preference for the sign of λ on these benchmarks: $\lambda = 1$ matches or exceeds $\lambda = -2/3$ on average for both models (LLaMA-3: 59.97 vs. 59.53; Mistral: 57.24 vs. 57.14), and the column-best is spread across methods and λ values. The per-λ differences (< 1 point) are within the seed variance we observe on the alignment benchmarks (Table 12) and should not be over-interpreted. Random and full-data DPO are marginally higher on average, which is expected: these benchmarks measure capabilities outside the objective our selection targets. In short, high likelihood ratio samples are the right choice for alignment (Fig. 9), and this choice comes at no cost to general capability.

Table 16: Performance comparison on general benchmarks (Llama-3-8B)

Model Name	GPQA	GSM8K	MMLU	Arc	TruthfulQA	HellaSwag	WinoGrande	Average
SFT	31.70	50.87	60.55	55.20	45.34	78.50	**73.48**	56.52
DPO	33.93	55.12	62.60	61.09	**57.39**	**81.37**	73.24	**60.68**
Random	**35.71**	55.95	62.44	60.75	55.95	80.83	72.30	60.56
Reward Margin	34.82	48.75	62.63	57.85	51.34	80.15	71.19	58.10
Ours ($\lambda = -1$)	35.27	52.08	62.54	**61.35**	54.53	81.31	73.01	60.01
Ours ($\lambda = -2/3$)	35.49	55.04	62.49	58.28	54.24	80.31	70.88	59.53
Ours ($\lambda = 2/3$)	34.60	51.55	62.69	58.36	54.61	80.26	71.90	59.14
Ours ($\lambda = 1$)	33.48	**56.10**	**62.95**	58.96	55.61	80.49	72.22	59.97

Table 17: Performance comparison on general benchmarks (Mistral-7B)

Model Name	GPQA	GSM8K	MMLU	Arc	TruthfulQA	HellaSwag	WinoGrande	Average
SFT	29.24	39.42	57.42	52.13	43.57	80.21	74.43	53.77
DPO	33.04	30.40	57.53	**60.15**	53.77	82.65	**75.61**	56.16
Random	33.04	**42.08**	58.42	59.47	64.25	81.26	73.56	**58.87**
Reward Margin	33.26	14.48	58.58	53.58	51.95	77.65	71.35	51.55
Ours ($\lambda = -1$)	30.80	30.25	54.58	58.62	59.63	**82.85**	72.77	55.64
Ours ($\lambda = -2/3$)	31.92	40.71	**58.82**	54.10	63.26	78.95	72.22	57.14
Ours ($\lambda = 2/3$)	**33.71**	39.80	56.72	49.66	**66.87**	79.67	72.30	56.96
Ours ($\lambda = 1$)	32.59	38.21	56.79	56.06	64.95	79.81	72.30	57.24

Performances: Instruct Checkpoint We report the model performances when the base models start from the instruction tuned versions as Table 18. Again, our method demonstrates superior data efficiency and absolute performance compared to both full dataset training and existing subset selection baselines.

Table 18: AlpacaEval2. We got the results from (Deng et al., 2025), only SPPO from (Wu et al., 2025). Numbers inside the parentheses are the percentage of selected samples.

Model	Mistral-7B-Instruct-v0.2		LLaMA3-8B-Instruct	
	LC (%)	WR (%)	LC (%)	WR (%)
SFT	17.11	14.72	22.92	22.57
DPO (Rafailov et al., 2023) (100%)	18.00	18.77	28.64	26.54
SPPO (Wu et al., 2025) (33%)	24.79	23.51	31.73	31.74
SPPO (Wu et al., 2025) (100%)	28.53	**31.02**	38.77	39.85
BeeS (Deng et al., 2025) (10%)	26.04	20.53	30.03	30.46
Ours (10%)	**30.13**	26.23	33.00	31.12
Ours (50%)	26.96	28.11	**41.69**	**42.03**

Performances: Code Domain To test whether our findings are specific to UltraFeedback, we train on a code preference dataset (Code-Preference-Pairs [8]) and evaluate code generation with HumanEval(+) and MBPP(+) under EvalPlus (Liu et al., 2023), reporting greedy pass@1 on the base and plus test suites. We use Qwen2.5-Coder-7B [9] and its instruct variant [10], comparing full-data DPO, Random, Margin, and Ours at a 50% selection ratio (Tables 19–20).

Two trends emerge. First, full-data DPO is harmful in this setting and substantially degrades both starting models, underscoring that selection is necessary rather than merely efficient. Second, on the base model, where there is room to move, the choice of criterion matters strongly: Ours retains 0.677/0.622 on HumanEval/HumanEval+ versus 0.463/0.427 for Random and 0.244/0.226 for Margin. On the instruct model, whose strong initialization leaves little headroom, all 50% subsets converge and Random is on par with or slightly above Ours on some metrics, while Ours remains above Margin on all four. The informative comparison is thus the base model, where likelihood ratio aware selection transfers to a substantially different preference domain.

Table 19: Code generation pass@1, Qwen2.5-Coder-7B (base).

	HumanEval	HumanEval+	MBPP	MBPP+
Qwen2.5-Coder-7B	0.835	0.793	0.796	0.688
Full data (100%)	0.652	0.598	0.772	0.643
Random (50%)	0.463	0.427	0.780	0.651
Margin (50%)	0.244	0.226	0.749	0.640
Ours (50%)	**0.677**	**0.622**	0.767	**0.656**

[8] https://huggingface.co/datasets/Vezora/Code-Preference-Pairs
[9] https://huggingface.co/Qwen/Qwen2.5-Coder-7B
[10] https://huggingface.co/Qwen/Qwen2.5-Coder-7B-Instruct

Table 20: Code generation pass@1, Qwen2.5-Coder-7B-Instruct.

	HumanEval	HumanEval+	MBPP	MBPP+
Qwen2.5-Coder-7B-Instruct	0.921	0.848	0.828	0.720
Full data (100%)	0.768	0.701	0.817	0.698
Random (50%)	**0.805**	**0.738**	**0.860**	**0.730**
Margin (50%)	0.756	0.683	0.825	0.709
Ours (50%)	0.799	0.720	0.841	**0.730**

Performances: Clean Training Data Table 21 is the performances when we change the training dataset for preference learning from Ultrafeedback-binarized to Argilla-7K (Argilla, 2024). Argilla-7k is a human curated, high quality yet small dataset. For a small dataset, sample selection would lead to performance degradation. However, our method (50% selection) maintains a competitive performance with the full dataset, meaning our method effectively identifies informative samples. We also confirmed high LLR samples remain effective for preference learning even in clean domains.

Table 21: Performances on Argilla Dataset (LLaMA-3-8B)

Model	Selection (%)	WR (%)	LC (%)
Llama3-8B	100	6.75	**12.43**
Random Sampling	50	4.15	7.68
Selection by loss		3.32	7.04
Ours ($\lambda = -2/3$)		4.06	8.46
Ours ($\lambda = 2/3$)		**6.92**	9.96

Performances: Iterative Method Table 22 compares the performances with iterative method, meaning using only subset of samples for one training iteration and repeat multiple iterations with different subsets.

The performances of DPO or IPO does not improve when iter 2 $\rightarrow$ iter 3, meaning the performances are converged. This convergence suggests that the sheer quantity of data processed through multiple iterations is less critical than the fundamental quality and informativeness of the training samples.

As shown in the results, our proposed strategy achieves superior or highly competitive performance despite requiring only a single iteration of training. In the case of LLaMA-3-8B-Instruct, our method records an LC Win Rate of 41.69%, significantly surpassing the performance of SPPO at Iteration 3 (38.77%). For Mistral-7B-Instruct-v0.2, our approach achieves a peak LC Win Rate of 30.13%, outperforming the most robust iterative baseline, SPPO Iter 3 (28.53%).

Furthermore, our method maintains a more efficient average response length (1682 for Mistral) compared to iterative methods like SPPO (2163), which tend to increase verbosity as iterations progress. These results demonstrate that our likelihood ratio aware data selection effectively identifies high quality samples, eliminating the need for computationally expensive iterative training while achieving a higher performance ceiling.

Performances: Leaderboard In Table 23, we compare the performances of the model trained by our likelihood ratio aware selected samples on the AlpacaEval2.0 leaderboard with other state-of-the-art AI chatbots. Our model outperforms many competing models trained on proprietary alignment data. When applied to Llama3-8B-Instruct, our model is even competitive to Claude 3 Opus and Llama-3.1-405B-Instruct.

Extensibility to Online RL While our primary focus has been the offline preference alignment setting, the proposed likelihood aware data selection framework is fundamentally applicable to online reinforcement learning (RL). To demonstrate this extensibility, we conduct experiments using GRPO (Shao et al., 2024), a popular online RL algorithm for LLM alignment.

Table 22: Performance comparison with iterative baselines. We got baseline results from (Wu et al., 2025).

Model	AlpacaEval 2.0 LC Win Rate	AlpacaEval 2.0 Win Rate	Avg. Len
Mistral-7B-Instruct-v0.2	17.11	14.72	1676
Snorkel (Mistral-PairRM-DPO)	26.39	30.22	2736
Self-Rewarding 70B Iter1	-	9.94	1092
Self-Rewarding 70B Iter2	-	15.38	1552
Self-Rewarding 70B Iter3	-	20.44	2552
Mistral-7B-DPO Iter1	23.81	20.44	1723
Mistral-7B-DPO Iter2	24.23	24.46	2028
Mistral-7B-DPO Iter3	22.30	23.39	2189
Mistral-7B-IPO Iter1	23.78	20.77	1693
Mistral-7B-IPO Iter2	21.08	23.38	2660
Mistral-7B-IPO Iter3	20.06	22.47	2760
Mistral-7B-SPPO Iter1	24.79	23.51	1855
Mistral-7B-SPPO Iter2	26.89	27.62	2019
Mistral-7B-SPPO Iter3	28.53	**31.02**	2163
Ours	**30.13**	28.10	1682
Llama-3-8B-Instruct	22.92	22.57	1899
Llama-3-8B-SPPO Iter1	31.73	31.74	1962
Llama-3-8B-SPPO Iter2	35.15	35.98	2021
Llama-3-8B-SPPO Iter3	38.77	39.85	2066
Ours	**41.69**	**42.03**	2053

Table 23: Performance comparison with leaderboard

Model	AlpacaEval 2.0 LC Win Rate	AlpacaEval 2.0 Win Rate
GPT-4o	57.5	51.3
GPT-4 Turbo	50.0	50.0
Ours (Llama-3-8B-Instruct)	**41.7**	**42.0**
Claude 3 Opus	40.5	29.1
Llama-3.1-405B-Instruct	39.3	39.1
GPT-4 0314	35.3	22.1
Claude-3-sonnet	34.9	25.6
Llama-3-70B-Instruct	34.4	33.2
Mistral Large	32.7	21.4
GPT-4 0613	30.2	15.8
Ours (Mistral-7B-Instruct-v0.2)	**30.1**	**28.1**
Mistral Medium	28.6	21.9
Gemini Pro	24.4	18.2
Llama-3-8B-Instruct	22.9	22.6
Mistral-7B-Instruct-v0.2	17.1	14.7

In the online setting, responses are generated dynamically at each training iteration. Therefore, quality evaluation and sample selection must be performed on-the-fly before the gradient update. Figure 17 illustrates the training dynamics of vanilla GRPO compared to two variants with selection: selection by an external reward model and selection with the likelihood ratio.

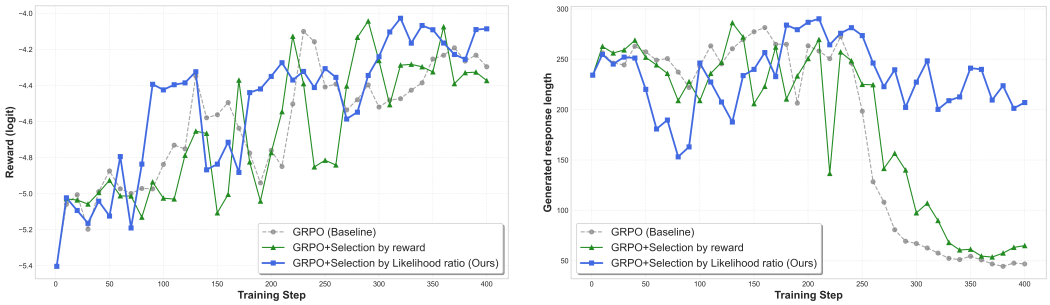


Figure 17: GRPO (a) reward and (b) response length curve while training.

Notably, vanilla GRPO suffers from significant length collapse, where the policy learns to exploit rewards by generating increasingly shorter and less informative outputs. While such collapse is often attributed to hyperparameter sensitivity, our results suggest that selection by likelihood ratio acts as a robust distributional regularizer, effectively preventing the model from drifting into these suboptimal, collapsed regions.

For evaluation, we report the AlpacaEval 2.0 performance and the various quality metrics as follows.

Table 24: Comparison of GRPO variants. Bold values indicate the best performance.

Method	LC	Win Rate	Avg. Token	Self-BLEU(↓)	Entity Density	Entropy
GRPO	5.11	2.39	167.90	0.16	**1.70**	**9.89**
+Selection by reward	4.29	2.20	177.53	0.15	1.56	9.82
+Selection by likelihood ratio	**5.41**	**2.96**	**179.31**	**0.14**	1.60	9.86

When the policy is optimized using responses filtered by their likelihood ratio, it achieves the highest Win Rate and LC Win Rate. More importantly, it yields the lowest Self-BLEU (0.14), indicating that our method successfully reduces structural redundancy and repetitive patterns. Despite the prevention of length collapse (evidenced by the highest Average Token length), the model maintains its informativeness, as verified by competitive Entity Density and Entropy scores. These findings confirm that likelihood ratio aware selection captures an objective-agnostic data value that transcends specific learning frameworks, providing stability even in iterative, online optimization environments.

D Limitations

Although our method reduces total training time compared to full dataset DPO and we showed it is necessary step for selecting high quality samples, π_θ estimation step takes time and it can be a computational bottleneck. Furthermore, as the likelihood ratio depends on the reference model, the quality of initial SFT model is important. Theorem 3.1 is stated for the optimal policy, whereas the implemented score uses a DPO-trained proxy (Remark 3.2); our evidence for the proxy's adequacy is empirical. Finally, variance over training seeds is reported for LLaMA-3 only (Table 12) due to the cost of repeated full fine-tuning.

E Future Works

Future research could explore adaptive scheduling of λ during training, and two-stage or nonlinear selection criteria beyond the linear family studied here (Appendix C.4). Furthermore, as our cur-

rent framework operates mainly in an offline setting, extending likelihood ratio awareness to online preference learning could lead to more real time alignment.

F BROADER IMPACT

This paper presents work whose goal is to advance the field of Machine Learning by improving the efficiency and reliability of preference alignment in Large Language Models. By proposing a data selection framework that prioritizes high value samples, our work reduces the computational resources required for model training, thereby mitigating the environmental impact of large-scale AI development. Furthermore, our likelihood-ratio-aware approach enhances training stability and provides a more principled foundation for aligning AI systems with human values, contributing to the development of safer and more robust AI assistants.