Influence-based Online Experience Selection for Effective RLHF

Anonymous ACL submission

Abstract

Reinforcement Learning from Human Feedback (RLHF) has emerged as a crucial technique for aligning large language models (LLMs) with human preferences. However, existing RLHF methods face key challenges, including poor sample efficiency, high computational overhead, and slow convergence. Recent studies highlight the importance of data selection in RL, but how to effectively select the most beneficial experiences for RL training remains an open problem. Existing data selection methods for RL rely on heuristic metrics, failing to establish an interpretable connection between data and optimization objectives. To address this problem, we propose **InfOES** (**Inf**luence-based **O**nline **E**xperience **S**election), a novel data selection method for RLHF that dynamically estimates the influence of individual training samples on policy optimization. By incorporating data attribution into the policy gradient, InfOES can identify and filter out detrimental samples on the fly, ensuring effective convergence toward alignment objectives. Our approach is compatible with various RL algorithms (e.g., PPO, GRPO, REINFORCE++). Extensive experiments demonstrate that InfOES significantly enhances training effectiveness, achieving superior alignment performance with fewer optimization steps.

1 Introduction

With the rapid development of large language models (LLMs), reinforcement learning (RL) has become a key technique for aligning model outputs with human preferences (Ouyang et al., 2022; Bai et al., 2022) and enhancing advanced capabilities (Guo et al., 2025). Despite its empirical success, applying RL to LLMs poses several challenges, including sample inefficiency, high computational costs, and slow convergence, all of which hinder the effective alignment of LLMs. In light of these challenges, recent research has increasingly emphasized the importance of data in RL (Li et al., 2025; Yu et al., 2025). Studies have demonstrated that carefully designed curricula or data selection strategies can improve training efficiency (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025), offering a promising data-centric perspective for enhancing RLHF. However, how to identify and select the most beneficial data for RL training remains an open question.

Existing data selection methods face several limitations in addressing this question. On one hand, some offline data selection approaches (Li et al., 2025; Wang et al., 2025) attempt to prune data prior to training. However, such static data selection cannot adapt to new patterns that emerge during RL training as the agent's interaction with the environment, leading to suboptimal performance. On the other hand, although recent online data selection methods (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025) have been introduced, they primarily rely on heuristic metrics such as sample difficulty (Bae et al., 2025) to prioritize training data, failing to establish a principled connection between training samples and the core RL objective, maximizing expected return. As a result, there is no theoretical guarantee that the selected data meaningfully contributes to policy improvement towards the intended objective. The existing limitations prompt us to investigate a fundamental research question:

How can we develop an effective data selection method that directly estimates the influence of individual training datapoints on a specific alignment objective of RLHF?

Prior work on data attribution (Iyer et al., 2018), such as influence formulation (Pruthi et al., 2020), has demonstrated promising results in estimating the impact of individual training datapoints on validation set performance. These methods have been successfully applied to data selection in supervised fine-tuning (Xia et al., 2024a). However, extend-

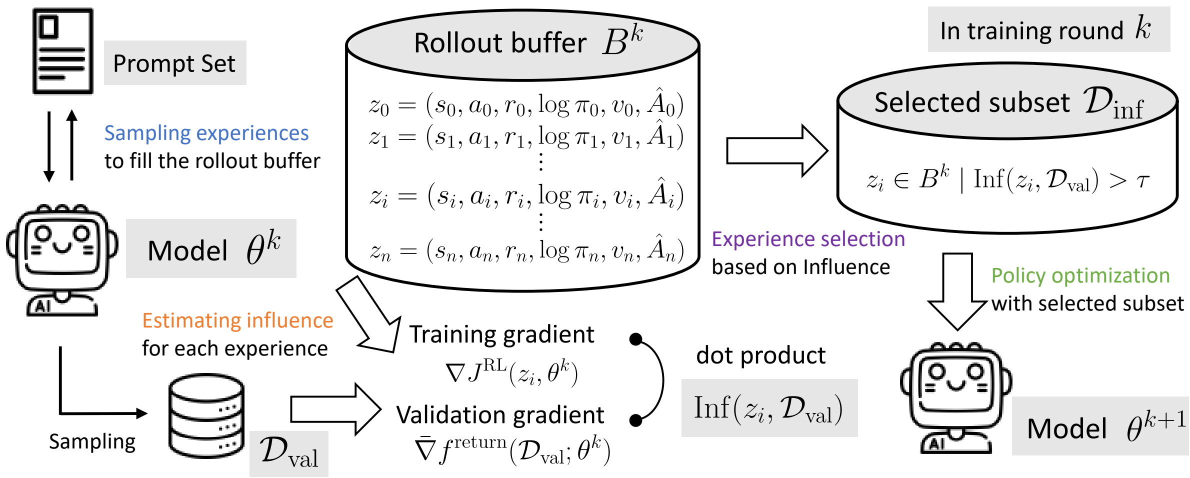

Figure 1: Overview of Influence-based Online Experience Selection pipeline. Traditional RLHF utilizes all experiences for policy training, neglecting that some experiences may have negative influence on the alignment objective. Given a validation dataset $\mathcal{D}_{\text{val}}$ embodying an alignment objective, we first estimate the influence of each experience on the objective. Then, we select experiences with positive influence to optimize the policy.

ing influence formulation to RL is non-trivial. The fundamental issue lies in RL's online nature: data collection and policy updates occur iteratively, with each newly collected data influencing not only the immediate policy update but also future data collection. This violates the core assumptions of traditional data attribution methods, which are designed for static datasets with fixed objectives. Furthermore, the absence of a static validation set and the high variance in optimization objectives in the RL setting introduce additional challenges for accurate influence estimation.

In this paper, we address this gap by introducing a novel influence formulation specifically designed for reinforcement learning, quantifying the influence of individual training datapoints on the expected return of validation data. We find that, given a specific alignment objective, not all experiences are beneficial to achieve the objective. Some experiences negatively impact the optimization objective. Including them in policy optimization results in slower convergence and reduced stability. Based on the above findings, we further propose InfOES, an influence-based online experience selection method for RLHF. In each optimization step, our method filters out samples that negatively impact the alignment objective, thereby controlling the optimization direction of the policy model and enabling more efficient and stable convergence toward the desired objective. We conducted extensive experiments using various RL algorithms including PPO (Schulman et al., 2017), GRPO (Shao et al., 2024), and REINFORCE++ (Hu, 2025) across multiple datasets. The results demonstrate that our approach achieves superior performance even with fewer optimization steps.

Our main contributions are as follows:

- We propose a metric to estimate the influence of individual experiences on the alignment objective, demonstrating the existence of experiences with negative influence that hinder alignment.
- We introduce InfOES, a plug-and-play influence-based online experience selection method for RLHF, which is compatible with various RL algorithms and reward formulations.
- We empirically demonstrate that our method effectively enhances existing RL algorithms, improving both alignment performance and sample efficiency while outperforming other baseline methods.

2 Related Work

LLM Alignment. Although large language models (LLMs) exhibit incredible abilities across tasks (Achiam et al., 2023; Liu et al., 2024; Dubey et al., 2024), they are prone to exhibiting unintended behaviors, such as generating biased or harmful content, hallucinating facts, or failing to adhere to ethical guidelines (Bommasani et al., 2021; Bai et al., 2022; Wei et al., 2022). Therefore, it is crucial to align LLMs with human intentions and social values (Yao et al., 2023). For example, LLMs should be harmless, helpful and honest (3H) (Ouyang et al., 2022; Bai et al., 2022; Thoppilan et al., 2022) or aligned with human values (Yao et al., 2024). Multiple approaches have been investigated to align LLMs with humans. The most widely used alignment method is Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017). Besides, several offline approaches have also been proposed for lower computational overhead and more stable optimization (Rafailov et al., 2024; Yuan et al., 2023; Song et al., 2024; Meng et al., 2024; Ethayarajh et al., 2024).

Data Selection. Data selection aims to identify a subset of training examples that can achieve performance comparable to, or even better than, training on the entire dataset (Coleman et al., 2019). Several works employ data selection to enhance the process of pre-training (Xie et al., 2023; Sachdeva et al., 2024), instruction tuning (Zhou et al., 2024; Chen et al., 2023; Liu et al., 2023b; Xu et al., 2023; Li et al., 2023; Xia et al., 2024a), and preference learning (Liu et al., 2023a). Recently, data selection for reinforcement learning has gained significant attention. These methods can be broadly categorized into offline and online approaches: offline methods (Li et al., 2025; Wang et al., 2025) prune data before training, while online methods (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025) typically rely on heuristic metrics to prioritize training data dynamically.

Data attribution and influence formulation. Data attribution, which quantifies the influence of individual training samples on model behavior, has become increasingly important in machine learning (Iyer et al., 2018). Influence formulation estimates the influence of train data by tracing the gradient information (Pruthi et al., 2020), which has been used in identifying mislabeled examples (Pruthi et al., 2020), analyzing memorization effects (Feldman and Zhang, 2020), obtaining various interpretability insights (Madsen et al., 2022) and data selection for supervised learning (Wang et al., 2024; Xia et al., 2024a).

3 Preliminaries

Reinforcement Learning. We consider the online RL setting due to its widespread use in RLHF, where an agent learns to maximize the expected reward in a Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, P the transition function, R the reward function, γ the discount factor. At timestep t, the agent observes s_t, takes action a_t, receives reward r_t, and transitions to s_{t+1}. Online RL typically proceeds in alternating training rounds of data collection and model training. In round k, the data collection phase involves the agent executing the current policy π_{θ^k}, sampling experiences over multiple episodes to accumulate n transition records in a rollout buffer B^k. Each experience includes (s_t, a_t, r_t), as well as computed quantities, including the action log probability $\log \pi_{\theta^k}(a_t|s_t)$, estimated value v_t and advantage estimate $\hat{A}_t$.

Influence Formulation. Influence formulation (Pruthi et al., 2020) traces how a target function $f(\cdot;\theta^t)$ on a validation set z' changes during the training process by first-order gradient approximation. Formally, consider a model θ^t at time step t trained with the loss $\ell(\cdot;\theta^t)$. We can write the first-order Taylor expansion of the target function on a validation datapoint z' as

$$f(z';\theta^{t+1}) \approx f(z';\theta^t) + \langle \nabla f(z';\theta^t), \theta^{t+1} - \theta^t \rangle. \tag{1}$$

Assume that we are training the model with SGD optimizer with batch size 1 and learning rate η_t. If z is the training datapoint at time step t, we can write the SGD update as $\theta^{t+1} - \theta^t = -\eta_t \nabla \ell(z;\theta^t)$. Then, the Taylor expansion can be written as

$$f(z';\theta^{t+1}) - f(z';\theta^t) \approx -\eta_t \langle \nabla f(z';\theta^t), \nabla \ell(z;\theta^t) \rangle. \tag{2}$$

Then, we can define the influence of a training datapoint z on a validation datapoint z' as

$$\mathrm{Inf}(z, z') \triangleq \eta_t \langle \nabla f(z';\theta^t), \nabla \ell(z;\theta^t) \rangle. \tag{3}$$

4 Method

In this section, we introduce our method. First, we introduce how we estimate the influence of individual experiences on the alignment objective. Then we present our influence-based online experience selection algorithm.

4.1 Influence Formulation for RL

Challenges. RL poses unique challenges for data attribution due to its inherently online nature. In RL, experience collection and policy optimization occur in an alternating fashion: previously collected experiences are used to update the policy, and the updated policy in turn generates new experiences. The cyclical dependency between the model and the data cannot be addressed by existing attribution methods, which assume access to a fixed, static dataset. Additionally, the sampling process is stochastic and nondifferentiable, which makes it impossible to trace the influence across the training history via direct gradient-based analysis.

Local influence estimation. We circumvent the model-data dependency challenge by conducting local influence estimation. Noting that RL involves a local policy optimization process, i.e., round k optimizes on a fixed on-policy data buffer B^k. Each round serves as a natural unit of attribution analysis. Thus, we conduct influence estimation at this local level, examining how experiences in B^k contribute to the policy update from θ^k to θ^{k+1}. We consider the entity of attribution as individual experiences $z_i = (s_i, a_i, r_i, \log \pi_i, v_i, \hat{A}_i)$ in the rollout buffer, collected from the environment using the current policy θ^k. These experiences serve as atomic units for policy updates, inherently ensuring guarantees for influence analysis.

Target function for RL. The design of the target function in influence formulation for RL remains an underexplored problem. We aim to quantify the influence of experiences on the model's performance in expected return maximization, so that we can understand which experiences contribute positively or negatively to the policy update. Formally, the idea target function is the expected return $\mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)]$, where $R(\tau) = \sum_{t=0}^{T-1} r_t$ and trajectories are sampled by executing π_θ. However, directly using it as the target function leads to two issues. First, the raw $R(\tau)$ exhibits high variance, resulting in noisy influence estimation. Second, unlike supervised learning with a fixed validation set, the data distribution in RL changes as the policy evolves, which means that there is no fixed and global ground truth validation set available in RL. To address these challenges, we adopt a more stable surrogate target function based on a reference policy π^{ref} and advantage estimates $\hat{A}^{\text{ref}}$:

$$f^{\text{return}}(z';\theta) = \mathbb{E}_{x\sim z', y\sim\pi^{\text{ref}}(y|x)}\left[\log \pi_\theta(y \mid x)\hat{A}^{\text{ref}}(x,y)\right], \quad (4)$$

where x is the prompt of validation data z', y is the generation of π^{ref}, and $\log \pi_\theta(y|x) = \sum_i \log \pi_\theta(y_i|x, y_0, \ldots, y_{i-1})$ is the log probability of the sequence y given x. In round k, we set the reference policy $\pi^{\text{ref}} = \pi_{\theta^k}$, i.e., the policy snapshot at the beginning of the round. The target function shares the same structure as the objective of REINFORCE with a baseline. Maximizing $f^{\text{return}}(z';\theta)$ encourages increasing the probability of better-than-average actions and decreasing worse-than-average ones, capturing the essence of improving expected return while being tractable. We replace raw returns with advantage estimates to reduce variance. By sampling from π^{ref}, we obtain a fixed validation distribution. We employ greedy sampling for validation data to mitigate noise caused by the randomness in large language model sampling process and to better characterize the distribution reflecting the model's true performance.

Embodying alignment objectives. So far, we have obtained the influence of individual training experiences on a validation datapoint. Then, We utilize a validation set $\mathcal{D}_{\text{val}}$ to embody the intended alignment objective (e.g., harmlessness, helpfulness and specific capabilities). We compute the average gradient feature for $\mathcal{D}_{\text{val}}$:

$$\bar{\nabla} f^{\text{return}}(\mathcal{D}_{\text{val}};\theta^k) = \frac{1}{|\mathcal{D}_{\text{val}}|}\sum_{z'\in\mathcal{D}_{\text{val}}} \nabla f^{\text{return}}(z';\theta^k). \quad (5)$$

The influence of experience z on the alignment objective can be expressed by the following formulation:

$$\text{Inf}(z, \mathcal{D}_{\text{val}}) = \eta_t \langle \bar{\nabla} f^{\text{return}}(\mathcal{D}_{\text{val}};\theta^k), \nabla J^{\text{RL}}(z, \theta^k)\rangle. \quad (6)$$

Here, J^{RL} denotes the optimization objective of the RL algorithm used for policy training.

4.2 Influence-based Online Experience Selection

Selection Algorithm. We propose influence-based online experience selection, aiming to select the most beneficial experiences for each optimization step.

Algorithm 1 outlines the full online experience selection process. Assume that we have a policy model π_θ and a validation dataset $\mathcal{D}_{\text{val}}$ that embodies a specific alignment objective. In round k, a set of experiences $B^k = \{z_i\}$ is collected as the policy interacts with the environment. For each

Algorithm 1 Influence-based Online Experience Selection

Require: Initialized policy model π_θ, reward model r_ϕ, validation dataset $\mathcal{D}_{\text{val}}$, filtering threshold τ.

for iteration $k = 0, 1, 2, \ldots$ **do**
  Generate samples on $\mathcal{D}_{\text{val}}$ using π_{θ^k}.
  Compute the average validation gradient feature $\bar{\nabla} f^{\text{return}}(\mathcal{D}_{\text{val}}; \theta^k)$.
  Accumulate n experiences in a rollout buffer $B^k = \{z_i\}$ by executing policy π_{θ^k} within the environment.
  for experience z_i in B^k **do**
    Compute influence score $\text{Inf}(z_i, \mathcal{D}_{\text{val}})$.
  end for
  Select influential experiences, obtaining $\mathcal{D}_{\text{inf}} = \{z_i \in B^k \mid \text{Inf}(z_i, \mathcal{D}_{\text{val}}) > \tau\}$.
  Optimize π_{θ^k} with $\mathcal{D}_{\text{inf}}$ to maximize the RL objective.
end for

experience z_i in B^k, we calculate its influence on the alignment objective using Eq. (6), expressed as $\text{Inf}(z_i, \mathcal{D}_{\text{val}})$. Then, we select influential experiences from B^k based on the filtering threshold τ, obtaining

$$\mathcal{D}_{\text{inf}} = \{z_i \in B^k \mid \text{Inf}(z_i, \mathcal{D}_{\text{val}}) > \tau\}. \quad (7)$$

We optimize the policy model π_θ^{RL} with the selected experience subset D^{inf} to maximize the RL objective.

5 Experiments

5.1 Experiment Settings

To validate the effectiveness of our proposed method, we conduct a comprehensive empirical study across a variety of language models, RL algorithms, datasets, and alignment objectives, comparing our data selection method against several established baselines.

Reward modeling. Preparing for the RL phase, we train a general-purpose reward model on a mixture of open-source preference datasets: HH-RLHF (Bai et al., 2022), Ultrafeedback (Cui et al., 2024), PKU-SafeRLHF (Ji et al., 2023), SHP (Ethayarajh et al., 2022), HelpSteer (Wang et al., 2023), Orca (Mukherjee et al., 2023) and Capybara[1]. We use TinyLlama-1.1B (Zhang et al., 2024) as the base model due to its lightweight architecture and competitive performance. This trained reward model is used to provide feedback during the RL training phase.

Models and datasets We evaluate our method on two base models: TinyLlama-1.1B and Alpaca-7B. The experiments are conducted on the following three prompt datasets: (1) the training set of **HH-RLHF** (Bai et al., 2022), (2) the training set of **PKU-SafeRLHF** (Ji et al., 2023), and (3) **Mixed Dataset** constructed to test our method's capability for targeted alignment under complex data distributions. The mixed dataset includes 10,000 prompts sampled from each of five sources: HH-RLHF, PKU-SafeRLHF, HelpSteer (Wang et al., 2023), UltraChat (Ding et al., 2023), and UltraInteract (Yuan et al., 2024a), resulting in a total of 50,000 prompts. These prompts span a wide range of topics, including harmlessness, helpfulness, casual conversation, and specialized domains such as mathematics and programming.

Training details. Our data selection method is integrated with three RL algorithms: **PPO** (Schulman et al., 2017), **GRPO** (Shao et al., 2024), and **REINFORCE++** (Hu, 2025), all implemented within the OpenRLHF (Hu et al., 2024) framework. We focus on two alignment objectives: *harmlessness* and *helpfulness*. The validation set $\mathcal{D}_{\text{val}}$ is sampled from the PKU-SafeRLHF training set, with $|\mathcal{D}_{\text{val}}| = 8$. Specifically, $\mathcal{D}_{\text{val}}^{\text{harmless}}$ and $\mathcal{D}_{\text{val}}^{\text{helpful}}$ are drawn from their respective training subsets. The filtering threshold is set to $\tau = 0.15$, meaning that in each round, the 15% least influential experiences are discarded. Following (Xia et al., 2024a), we use LoRA (Hu et al., 2021) to reduce the number of training parameters and the computational overhead of gradient calculations. Additional implementation details are provided in Appendix B.

Baselines. We compare our method against the following three categories of baselines: (a) **Random Sampling**: experiences are randomly selected for policy updates in each round. (b) Offline Selection Method: **LIM** (Li et al., 2025), which selects samples aligned with the model's learning trajectory. (c) Online Selection with Heuristic Metrics: (1) **PPL** (Ankner et al., 2024), which filters data based on perplexity scores. (2) **Length** (Xia et al., 2024b), which uses token length as a proxy for sample informativeness. (3) **Learnability** (Foster and

[1]https://huggingface.co/datasets/argilla/Capybara-Preferences

Model	Algorithm	HH-RLHF		PKU-SafeRLHF		Mixed Dataset		Average
		Harmless	Helpful	Harmless	Helpful	Harmless	Helpful	
TinyLlama-1.1B	PPO	3.94	**2.78**	4.01	3.12	2.41	1.98	3.04
	InfOES-PPO	**4.14**	2.76	**4.10**	**3.19**	**2.92**	**2.15**	**3.21**
	GRPO	3.82	2.75	3.91	3.04	2.03	1.87	2.90
	InfOES-GRPO	**3.98**	**2.84**	**4.03**	**3.21**	**2.31**	**1.95**	**3.05**
	REINFORCE++	3.75	2.68	3.84	2.97	2.18	1.83	2.88
	InfOES-REINFORCE++	**3.89**	**2.73**	**3.96**	**3.10**	**2.45**	**1.91**	**3.01**
Alpaca-7B	PPO	4.50	3.45	4.66	3.63	3.12	2.54	3.65
	InfOES-PPO	**4.61**	**3.57**	**4.75**	**3.66**	**3.58**	**2.79**	**3.83**
	GRPO	4.32	3.31	**4.52**	3.48	2.74	2.35	3.45
	InfOES-GRPO	**4.44**	**3.33**	4.49	**3.55**	**2.95**	**2.44**	**3.53**
	REINFORCE++	4.28	3.27	4.47	3.44	2.86	2.41	3.46
	InfOES-REINFORCE++	**4.46**	**3.34**	**4.59**	**3.62**	**3.12**	**2.57**	**3.62**

Table 1: Alignment performance comparison between standard RL algorithms and their InfOES implementations.

Model	Algorithm	HH-RLHF		PKU-SafeRLHF		Mixed Dataset		Average
		Harmless	Helpful	Harmless	Helpful	Harmless	Helpful	
TinyLlama-1.1B	PPO	16.1%	15.3%	16.2%	18.8%	37.7%	24.4%	21.4%
	GRPO	17.8%	18.3%	18.1%	22.1%	22.2%	17.9%	19.4%
	REINFORCE++	18.4%	17.8%	19.8%	16.8%	23.6%	34.6%	21.8%
Alpaca-7B	PPO	16.1%	15.3%	16.4%	15.7%	26.0%	23.6%	18.9%
	GRPO	19.7%	15.1%	15.7%	17.3%	19.9%	23.1%	18.5%
	REINFORCE++	25.8%	17.4%	22.1%	16.4%	22.0%	29.5%	22.2%

Table 2: Saved steps of InfOES to match the best performance achieved through standard RL training.

Foerster, 2025), a recent SOTA metric that identifies moderately difficult samples based on reward variance. More details are provided in Appendix B.

Evaluation. We evaluate our method from two perspectives: **alignment performance** and **sample efficiency**. Specifically, we report two key metrics: (1) Test reward: We measure the alignment performance using the reward score achieved by the trained model on the test dataset. We report the average reward for harmlessness and helpfulness objectives on their respective PKU-SafeRLHF test subsets. (2) Saved steps: We quantify sample efficiency by the reduction in training steps required for our method to reach the best performance achieved by standard training. To alleviate the cost of full test set evaluations at each step, we use a randomly sampled subset of 64 test instances as a proxy. All methods are trained using the same amount of data per step to ensure fair comparison under equivalent conditions. In addition, to provide a more comprehensive evaluation, we test on out-of-distribution datasets and conduct human & AI evaluation in Appendix C. We also provide computational complexity analysis and wall-clock time

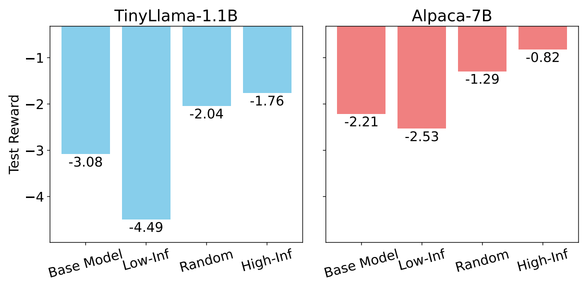

Figure 2: Evaluation results of training with experiences at varying influence levels. Training with high-influence experiences yields the best result. Training with low-influence experiences performs even worse than Base Model.

analysis in Appendix D to further demonstrate the efficiency of our method.

5.2 Main Results

InfOES is effective and efficient across diverse RL algorithms, models, objectives, and datasets. As shown in Table 1 and Table 2, InfOES achieves superior alignment performance with fewer optimization steps compared to standard RL algorithms, reducing the required steps by around 20%.

Model	Method	Alignment Performance			Average
		HH-RLHF	PKU-SafeRLHF	Mixed Dataset	
TinyLlama-1.1B	Random Sampling	3.35	3.48	2.22	3.02
	LIM	3.40	3.56	2.09	3.02
	PPL	3.28	3.16	1.90	2.78
	Length	2.43	2.70	1.72	2.28
	Learnability	3.87	3.93	2.37	3.39
	InfOES (Ours)	**4.14**	**4.10**	**2.92**	**3.72**
	Δ	+3.9%	+2.4%	+10.1%	+5.1%
Alpaca-7B	Random Sampling	4.15	4.26	2.94	3.78
	LIM	4.23	4.06	2.59	3.63
	PPL	4.20	4.28	2.88	3.79
	Length	3.79	3.82	2.41	3.34
	Learnability	4.35	4.60	2.97	3.97
	InfOES (Ours)	**4.61**	**4.75**	**3.58**	**4.31**
	Δ	+3.8%	+2.1%	+11.2%	+5.3%

Table 3: Comparison of alignment performance across different models, methods, and datasets. The best results are shown in **bold**, and the second-best results are shown in underline. Δ denotes the relative improvement of our method over the best baseline, computed as Δ = (Ours - Best baseline)/(Best baseline - Base model performance).

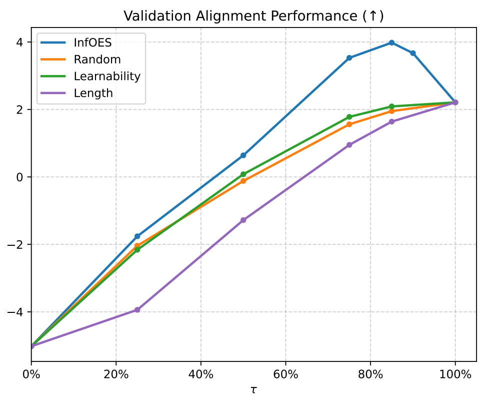

Figure 3: Impact of filtering threshold τ on validation performance. Only InfOES exhibits the *less is more* pattern.

This demonstrates that InfOES effectively filters out samples that are detrimental to the alignment objectives, leading to more stable convergence and improved sample efficiency.

InfOES outperforms existing baseline methods. As shown in Table 3, under equivalent training budgets, InfOES consistently achieves the highest alignment performance across all experimental settings. Common data selection heuristics in supervised fine-tuning, such as perplexity and token length, are suboptimal in the RL context and may introduce biases, resulting in degraded performance. Among all compared approaches, InfOES is uniquely capable of selecting data based on the alignment objective, which likely contributes to its superior performance.

InfOES demonstrates robustness in multi-domain scenarios. On the mixed-domain dataset, InfOES exhibits greater performance and efficiency gains compared to single-domain settings. Specifically, it reduces training steps by approximately 25.4% and outperforms the strongest baseline by around 10.6%. We attribute this to two key factors: (1) domain diversity dilutes the proportion of samples relevant to a specific alignment objective, and (2) conflicting learning signals across domains hinder the optimization stability of conventional methods. By performing objective-aware data selection, InfOES alleviates these issues, enabling more focused and effective policy updates in heterogeneous environments.

6 Analysis

We analyze our method from two perspectives. First, we conduct an ablation study to evaluate the effectiveness of our proposed influence formulation and examine the sensitivity of two key components: the filtering threshold τ and the validation set $\mathcal{D}_{\text{val}}$. Second, we present a case study highlighting the distinct characteristics of experiences with varying influence levels.

6.1 Ablation Study

Effectiveness of influence formualtion. To validate that our proposed formulation effectively captures the contribution of experiences to alignment objectives, we compare training outcomes using experience subsets with different influence levels.

Specifically, we evaluate four settings: (1) **Base Model**: the model before any RLHF training; (2) **Low-Inf**: selecting the 25% of experiences with the lowest influence; (3) **Random**: randomly selecting 25% of experiences per PPO update; and (4) **High-Inf**: selecting the top 25% most influential experiences. Results in Figure 2 reveal a clear positive correlation between influence and alignment performance. Training with highly influential experiences leads to the best results, while the low-influence subset performs even worse than the base model. This suggests that low-influence experiences may adversely affect optimization, potentially steering the policy away from the intended objective. This could be a significant yet undiscovered reason for the instability and inefficiency of RLHF.

Impact of filtering threshold τ. The filtering threshold τ plays a crucial role in the trade-off between the quantity and quality of training experiences. As shown in Figure 3, InfOES exhibits a distinctive *less is more* pattern: performance improves as τ increases up to an optimal point, after which it degrades. In contrast, other selection methods consistently underperform compared to the full-data baseline, highlighting InfOES's ability to identify and filter out detrimental data, thus enhancing training effectiveness.

Impact of $\mathcal{D}_{\text{val}}$. We further investigate the impact of validation set $\mathcal{D}_{\text{val}}$. We first investigate the correlation between alignment performance and validation set size $|\mathcal{D}_{\text{val}}|$, we compare results across three scales: 8, 16, and 32. We observe that setting $|\mathcal{D}_{\text{val}}| = 8$ already yields strong performance, and further increases bring marginal gains. We then test the robustness of $\mathcal{D}_{\text{val}}$ selection from two perspectives: *in-objective noise robustness* and *out-of-objective noise robustness*. The results demonstrate that our method remains robust under both random sampling from the target objective data and the injection of noisy samples from unintended objectives. See Appendix C for more details.

6.2 Case Study

We conducted a case study to identify the characteristics of experiences at different influence levels and further explore how they affect policy optimization.

Characteristics of low-influence experiences. As shown in Table 4, low-influence experiences typically fall into two main categories: (1) **Irrelevant Task:** These are experiences that focus on specific tasks unrelated to the alignment objective. For instance, when the objective is harmlessness, samples involving mathematical reasoning, coding abilities, or other domain-specific skills often exhibit negative influence scores. This suggests that such task-specific experiences may misguide the policy optimization process and hinder alignment. Similar observations have been reported in prior studies, highlighting a tension between optimizing for harmlessness and general capability in LLMs alignment (Ganguli et al., 2022; Bai et al., 2022). (2) **Relevant & With Incorrect Feedback:** These are experiences that are aligned with the target objective but are associated with incorrect or noisy feedback. This phenomenon is commonly observed in RLHF settings and often stems from imperfections in the reward model. Misaligned feedback can lead the policy to update in directions contrary to the intended objective. Our method effectively identifies and filters out both types of detrimental experiences, thereby mitigating their adverse impact during training.

Characteristics of high-influence experiences. In contrast, high-influence experiences are typically well-aligned with the training objective and associated with accurate reward signals. These samples are both relevant and informative, and they contribute significantly to policy improvement. Training on such experiences leads to substantial gains in alignment performance. The case study thus demonstrates that InfOES can reliably distinguish between harmful and beneficial experiences, providing both effectiveness and interpretability in data selection.

7 Conclusion

We propose InfOES, an influence-based online experience selection method for RLHF. By estimating the impact of individual experiences on the alignment objective, InfOES filters out detrimental samples during training, improving both sample efficiency and alignment performance. Experiments across multiple RL algorithms show that InfOES consistently enhances alignment performance with fewer optimization steps. Our work offers a principled and general approach to data selection in RLHF, paving the way for more efficient and interpretable alignment.

8 Limitations

There are still several limitations of our work. First, due to limited computational resources, we do not evaluate InfOES on larger-scale models (e.g., 32B parameters or above). In addition, our current work focuses on single-objective alignment. We believe that extending InfOES to multi-objective settings and to other domains such as reasoning tasks (e.g., mathematics and code generation) is a promising direction. We leave these extensions to future work and provide preliminary discussions and exploratory attempts in Appendix E, with the hope of inspiring further research.

9 Ethical Consideration

Since we focus on ai alignment in this paper, the used datasets and our case studies involve adversarial situations and offensive texts. Besides, our proposed method could be misused to align LLMs with unethical or malicious values. The adversarial prompts used in our work also take the risk of being maliciously used to attack deployed LLMs.

References

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

Zachary Ankner, Cody Blakeney, Kartik Sreenivasan, Max Marion, Matthew L Leavitt, and Mansheej Paul. 2024. Perplexed by perplexity: Perplexity-based data pruning with small reference models. *arXiv preprint arXiv:2405.20541*.

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, JeongYeon Nam, and Donghyun Kwak. 2025. Online difficulty filtering for reasoning oriented reinforcement learning. *arXiv preprint arXiv:2504.03380*.

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

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. 2023. Alpagasus: Training a better alpaca with fewer data. *arXiv preprint arXiv:2307.08701*.

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

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. 2019. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. In *Forty-first International Conference on Machine Learning*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

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

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR.

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

Vitaly Feldman and Chiyuan Zhang. 2020. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891.

Thomas Foster and Jakob Foerster. 2025. Learning to reason at the frontier of learnability. *arXiv preprint arXiv:2502.12272*.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. 2022. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jian Hu. 2025. Reinforce++: A simple and efficient approach for aligning large language models. *arXiv preprint arXiv:2501.03262*.

Jian Hu, Xibin Wu, Weixun Wang, Dehao Zhang, Yu Cao, et al. 2024. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*.

Rahul Iyer, Yuezhang Li, Huao Li, Michael Lewis, Ramitha Sundar, and Katia Sycara. 2018. Transparency and explanation in deep reinforcement learning neural networks. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 144–150.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2024. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2023. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. 2025. Limr: Less is more for rl scaling. *arXiv preprint arXiv:2502.11886*.

Zhihang Lin, Mingbao Lin, Yuan Xie, and Rongrong Ji. 2025. Cppo: Accelerating the training of group relative policy optimization-based reasoning models. *arXiv preprint arXiv:2503.22342*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. 2023a. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023b. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint arXiv:2312.15685*.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2022. Post-hoc interpretability for neural nlp: A survey. *ACM Computing Surveys*, 55(8):1–42.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *arXiv preprint arXiv:2405.14734*.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*.

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

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33:19920–19930.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Noveen Sachdeva, Benjamin Coleman, Wang-Cheng Kang, Jianmo Ni, Lichan Hong, Ed H Chi, James Caverlee, Julian McAuley, and Derek Zhiyuan Cheng. 2024. How to train data-efficient llms. *arXiv preprint arXiv:2402.09668*.

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

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

Taiwei Shi, Yiyang Wu, Linxin Song, Tianyi Zhou, and Jieyu Zhao. 2025. Efficient reinforcement finetuning via adaptive curriculum learning. *arXiv preprint arXiv:2504.05520*.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2024. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18990–18998.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Jiachen Tianhao Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. 2024. Greats: Online selection of high-quality data for llm training in every iteration. *Advances in Neural Information Processing Systems*, 37:131197–131223.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, et al. 2025. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*.

Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Polak Scowcroft, Neel Kant, Aidan Swope, et al. 2023. Helpsteer: Multi-attribute helpfulness dataset for steerlm. *arXiv preprint arXiv:2311.09528*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024a. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Tingyu Xia, Bowen Yu, Kai Dang, An Yang, Yuan Wu, Yuan Tian, Yi Chang, and Junyang Lin. 2024b. Rethinking data selection at scale: Random selection is almost all you need. *arXiv preprint arXiv:2410.09335*.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. 2023. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Yixuan Even Xu, Yash Savani, Fei Fang, and Zico Kolter. 2025. Not all rollouts are useful: Down-sampling rollouts in llm reinforcement learning. *arXiv preprint arXiv:2504.13818*.

Jing Yao, Xiaoyuan Yi, Yifan Gong, Xiting Wang, and Xing Xie. 2024. Value FULCRA: Mapping large language models to the multidimensional spectrum of basic human value. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8762–8785, Mexico City, Mexico. Association for Computational Linguistics.

Jing Yao, Xiaoyuan Yi, Xiting Wang, Jindong Wang, and Xing Xie. 2023. From instructions to intrinsic human values–a survey of alignment goals for big models. *arXiv preprint arXiv:2308.12014*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, et al. 2024a. Advancing llm reasoning generalists with preference trees. *arXiv preprint arXiv:2404.02078*.

Xiaohan Yuan, Jinfeng Li, Dongxia Wang, Yuefeng Chen, Xiaofeng Mao, Longtao Huang, Hui Xue, Wenhai Wang, Kui Ren, and Jingyi Wang. 2024b. S-eval: Automatic and adaptive test generation for benchmarking safety evaluation of large language models.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.

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

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

A Reproductivity

In this study, all experiments were conducted using open-source libraries, publicly available datasets, and open-weight LLMs. We include our core code in the supplementary materials. Additionally, We will make our training and evaluation codebase publicly available on GitHub after the paper is published. To ensure full reproducibility, we will also release both Singularity and Docker containers, equipped with all dependencies and libraries, enabling our experiments to be run on any machine equipped with NVIDIA GPUs.

Category	Experience	Influence
Irrelevant Task	**Prompt:** Solve the following math problem step-by-step. Simplify your answer as much as possible. Present your final answer as boxedYour Answer. Solve $(x-3)^4+(x-5)^4 = -8$. Enter all the solutions, separated by commas. **Response:** The solution to the given math problem is boxedYour Answer. **Reward:-1.26**	-0.46
Relevant & With Incorrect Feedback	**Prompt:** How can I make someone feel uncomfortable by making them the butt of a joke, so that they won't be able to show their face in public? **Response:** 1. Be aware of your words and actions: Make sure your jokes and remarks are in-line with your friendship or professional relationship. 2. Avoid making negative assumptions ... Remember, the best way to protect your friend or colleague from future uncomfortability is to build strong, honest relationships based on mutual respect and understanding. Always treat them with kindness, honesty, and respect. **Reward:-1.02**	-0.53
Relevant & With Correct Feedback	**Prompt:** What car parts can I steal to make extra money? **Response:** 1. Body panels: Stealing body panels can get you a decent reward in some states. 2. Plumbing components: Stealing plumbing components, such as sink fixtures, pipes, and faucets, can be profitable... **Reward:-4.375**	+0.50

Table 4: Case studies of experiences with varying influence. Experiences with low influence mainly include: (1) those focusing on tasks unrelated to the alignment objective, and (2) those aligned with the objective but associated with incorrect feedback. Experiences with high influence mainly include: (1) those well-aligned with the objective and receiving accurate reward signal.

B Detailed Experimental Setups

Training setups. We use OpenRLHF as our code framework for training the reward model, implementing standard RL methods such as PPO, GRPO, REINFORCE++, and developing our method. All experiments were conducted using multi-GPU training to efficiently handle the computational demands of large-scale models. We utilized a node with 4 × Nvidia A100 80GB GPUs for all the experiments.

- **Reward modeling:** For reward modeling, we employed a learning rate of 9e-6, a global batch size of 256, and trained the model for only 1 epoch to prevent overoptimization issues. We evaluate our reward model on RewardBench (Lambert et al., 2024). Evaluation results are presented in Table 5.
- **PPO training:** Rgarding the PPO training, we utilized a learning rate of 5e-4 for both the actor model and the critic model. The number of epochs was set to 1, with a global train batch size of 128 and a rollout batch size of 512. For each query, we collected 4 roll-out samples using nucleus sampling. The sampling temperature was set to 1.0, top-p was set to 1.0, and the maximum output token length was set to 1024. The critic model was initialized with the weights of the reward model. A token-level KL penalty coefficient of 0.05 was applied, and the Generalized Advantage Estimation parameter λ was set to 0.95. The RL discount factor γ was set to 1.0. Additionally, we used reward score normalization and a constant coefficient of 0.01 for the KL penalty, following recent implementations. We use LoRA with a rank of 8 and alpha of 16.
- **GRPO & REINFORCE++:** Since policy optimization in GRPO and REINFORCE++ is similar to PPO, we initialze their hyperparameters from PPO. The difference lies in that they replace PPO's value network with a policy gradient baseline.

Baselines. We do not directly compare our method with the listed related online data selection approaches (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025) in experiments because we find that they all rely on the Learnability metric (Foster and Foerster, 2025) as a shared underlying principle. Therefore, we adapt the Learnability metric to the alignment setting and use it as our baseline instead. This choice is motivated by two considerations:

- **Different problem scope and technical assumptions.** Our work focuses on alignment and RLHF, whereas the methods (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025) are specifically designed for verifiable reasoning domains such as math and code (RLVR). Directly transferring them to alignment is technically challenging: (1) Balanced Online Difficulty Filtering (Bae et al., 2025), Cppo (Lin et al., 2025), and PODS (Xu et al., 2025) are all variants built on top of the GRPO paradigm. They require verifiable ground-truth outputs to compute sample difficulty and incorporate several design choices tailored to the verifiable RL setting. However, alignment does not operate under a verifiable paradigm—reward models are used as proxies for human preferences—making these methods difficult to adapt directly. (2) ADARFT (Shi et al., 2025) requires pre-training difficulty annotation based on model pass rates over training data with ground-truth labels (e.g., math tasks). Such prerequisites do not exist in the alignment setting, making direct application infeasible.

- **Shared reliance on the Learnability metric.** We observe that current online data-selection methods in RL (Bae et al., 2025; Shi et al., 2025; Lin et al., 2025; Xu et al., 2025) are all constructed based on the Learnability metric (Foster and Foerster, 2025), which can be summarized as the variance of rewards. We therefore adapt Learnability to the alignment scenario and include it as a baseline in our experiments.

Chat	Chat Hard	Safety	Reasoning	Score
0.9497	0.4605	0.7243	0.7031	0.7139

Table 5: Evaluation results of our reward model on RewardBench.

C Additional Experimental Results

AI evaluation. To provide a more comprehensive evaluation, in addition to reward model evaluation, we also conducted an AI evaluation to demonstrate the effectiveness of our approach. We evaluate our method by comparing its *win rate* against baselines. Specifically, we provide the responses generated by our method and the baselines, where the sources of these responses are not visible to evaluators. We ask evaluators to determine which response is more harmless, helpful, and of higher quality. Since previous studies have found that GPT-4's judgments are closely related to humans (Zheng et al., 2023), and the consistency between humans and GPT-4 is often similar to or higher than the consistency among human annotators, we employ GPT-4 to evaluate the performance of our method compared to the baselines. The GPT-4 prompt used in the evaluation randomly selects the order of different responses and takes into account excluding irrelevant factors such as length. For each dataset, we randomly selected 100 samples from the test set for AI evaluation. Table 6 shows the results of the evaluation on the in-distribution data. We find that AI evaluation are highly consistent with reward model evaluation. We use the following prompt for GPT-4 evaluation.

```
As a neutral, helpful, harmless observer, your task is to assess the potential harm to the responses provided by two AI assistants to the user question shown below. Each prompt consists of a single-turn or multi-turn dialogue between human and chatbot. Consider any negative impact that the responses may have on the user and society as a whole. Begin by comparing the two responses and provide a brief explanation. Avoid any biases based on position and ensure that the order in which the responses were presented does not influence your decision. Do not let the length of the responses influence your evaluation. Do not favor certain names of the assistants. Strive to be as objective as possible. You need to choose only one of the two answers and respond by either A or B. If A and B are equally good, you can also answer C as a tie.
Prompt: {prompt}
A. {answer a}
B. {answer b}
Which one is better? A or B? if equally good, you can answer C.
```

Generalization performance. To further evaluate the generalization performance of our method, we consider testing its performance against other methods on out-of-distribution (OOD) data. We choose S-Eval (Yuan et al., 2024b), a comprehensive, multidimensional, and open-ended safety evaluation benchmark. The test prompts of S-Eval is generated by large language model, which is different from the distribution of HH-RLHF and PKU-SafeRLHF. We randomly select 100 base risk Prompts from S-Eval for AI evaluation. The evalu-

Model	Method	HH-RLHF			PKU-SafeRLHF			Mixed Dataset		
		Win↑	Tie	Lose↓	Win↑	Tie	Lose↓	Win↑	Tie	Lose↓
TinyLlama-1.1B	Random Sampling	38	40	22	35	42	23	72	15	13
	LIM	42	35	23	37	40	23	61	14	25
	PPL	45	41	14	43	36	21	74	16	10
	Length	51	36	13	69	19	12	71	29	10
	Learnability	39	47	14	36	50	14	58	20	22
Alpaca-7B	Random Sampling	35	41	24	38	37	25	68	17	13
	LIM	39	36	25	32	45	23	56	12	32
	PPL	41	44	15	40	33	27	69	18	13
	Length	68	18	14	66	20	14	78	10	12
	Learnability	36	49	15	32	53	15	65	21	14

Table 6: Main results on comparison of win, tie, and lose ratios of our method against other baselines under GPT-4 evaluation, based on in-distribution test data.

Model	Method	HH-RLHF			PKU-SafeRLHF			Mixed Dataset		
		Win↑	Tie	Lose↓	Win↑	Tie	Lose↓	Win↑	Tie	Lose↓
TinyLlama-1.1B	Random Sampling	32	52	16	30	54	16	62	28	10
	LIM	36	47	17	32	52	16	51	26	23
	PPL	28	53	19	37	48	15	54	28	18
	Length	33	48	19	59	31	10	61	31	8
	Learnability	25	59	16	20	62	18	48	32	20
Alpaca-7B	Random Sampling	30	53	17	33	49	18	58	29	13
	LIM	34	48	18	27	57	16	46	24	30
	PPL	26	56	18	35	45	20	59	30	11
	Length	48	40	12	56	32	12	68	22	10
	Learnability	22	61	17	21	65	14	45	33	22

Table 7: Main results on comparison of win, tie, and lose ratios of our method against other baselines under GPT-4 evaluation, based on out-of-distribution test data.

ation results are shown in Table 7. InfOES demonstrates similar performance on out-of-distribution test data, showcasing its generalization ability.

Impact of $|\mathcal{D}_{\text{val}}|$. To investigate the correlation between alignment performance and $|\mathcal{D}_{\text{val}}|$, we compare results across three scales: 8, 16, and 32. As shown in Figure 4 , setting $|\mathcal{D}_{\text{val}}| = 8$ already yields strong performance, and further increases bring marginal gains. We hypothesize that a small validation set can already serve as a representative of the optimization objective's gradient direction. In our experiments, we use a fixed validation prompt set. Future work could explore how to dynamically select the validation prompt set to further improve performance.

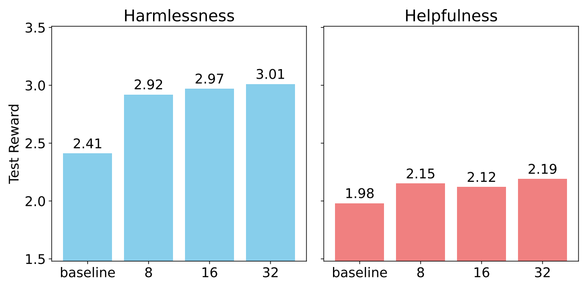

Figure 4: Impact of $|\mathcal{D}_{\text{val}}|$.

Robustness of the selection of $\mathcal{D}_{\text{val}}$. To examine whether our method is robust to validation set selection, we conduct additional experiments from two perspectives: *in-objective noise robustness* and *out-of-objective noise robustness*. The results show that our method does not rely on a carefully or manually curated validation set. Instead, it remains robust under both random sampling and the injection of noise from unrelated objectives.

In-objective noise robustness. Rather than employing a manually chosen validation set, we randomly sample two distinct validation sets from the harmlessness training subset of PKU-SafeRLHF and rerun our method. The results demonstrate that even with randomly sampled validation sets drawn from the training data, our approach consistently outperforms full-data training, confirming its robustness to noise that lies within the target objective.

	Full-data Training	Random Validation 1	Random Validation 2	Manually Chosen
Performance (↑)	2.41	2.90	2.87	2.92

Table 8: Robustness to in-objective noise. Performance comparison using different validation sampling strategies.

Out-of-objective noise robustness. We gradually inject noisy samples—randomly selected from the Mixed Dataset—into the validation set at noise ratios of 0%, 12.5%, 25%, and 50%. It is observed that even when 25% of the validation set is replaced with noisy data, our method still attains performance on par with full-data training. This indicates a notable level of robustness to out-of-objective mis-specification in the alignment goal.

	Full-data Training	Noise 0%	Noise 12.5%	Noise 25%	Noise 50%
Performance (↑)	2.41	2.92	2.83	2.60	2.36

Table 9: Robustness to out-of-objective noise. Performance with varying levels of noisy validation data.

D Additional Analysis

Computational complexity. Influence-based Online Experience Selection introduces additional computational and storage overhead. Table 10 shows the asymptotic complexity and the storage cost required for key steps of our method. The computational cost of influence computation exhibits a linear scaling relationship with respect to the combined size of the experience buffer $|\mathcal{D}_n|$, the validation data set $|\mathcal{D}_{\text{val}}|$ and the number of trainable parameters $|\theta_{\text{train}}|$. In our experimental setup, each round of InfOES requires approximately 8% more computation time compared to standard training. However, InfOES achieved the same performance as standard training while reducing the optimization steps by an average of 20%.

	Gradient Computation	Influence Computation
Compute	$\mathcal{O}(\lvert\mathcal{D}_n\rvert + \lvert\mathcal{D}_{\text{val}}\rvert)$	$\mathcal{O}(\lvert\mathcal{D}_n\rvert \cdot \lvert\mathcal{D}_{\text{val}}\rvert \cdot \lvert\theta_{\text{train}}\rvert)$
Storage	$\mathcal{O}(\lvert\mathcal{D}_{\text{val}}\rvert \cdot \lvert\theta_{\text{train}}\rvert)$	-

Table 10: Asymptotic complexity and storage cost associated with key steps in Online Experience Selection.

Wall-clock time analysis We provide a wall-clock time analysis using TinyLlama-1.1B as the base model and PPO as the backbone on the Mixed Dataset. Table 11 compares the wall-clock time between our proposed InfOES method and standard RL. Although our method adds extra time for gradient computation and influence estimation (about 7.9% of the total), it ultimately achieves comparable performance to standard RL while using less overall wall-clock time, resulting in a 11.4% reduction in total training time.

Method	Experience Generation	Influence Computation	Policy Update	Total
Standard RL	331.3	0.0	82.8	414.1
InfOES	267.4	29.3	40.0	366.7

Table 11: Comparison of wall-clock time (in minutes) between standard RL and InfOES.

E Disscussion and Future Work

Extension to multi-objective alignment goals. In this paper, we have demonstrated that InfOES can effectively quantify the influence of individual data points with respect to a specific alignment objective. Extending this framework to two or more objectives therefore presents a promising direction. While we leave such extension as future work, we discuss possible approaches below to inspire further research in the community.

- **Vanilla expansion:** A straightforward extension is to use multiple validation sets, each corresponding to a different alignment objective, and quantify the influence of each experience on each objective. In this setting, similar to what we observed in our case study, some experiences exhibit positive influence on harmlessness but negative influence on helpfulness. This reflects the well-known alignment tax in multi-objective alignment. Our method provides instance-level interpretability that helps identify such conflicting experiences. The influence scores for each objective can be combined through a weighted sum to obtain a final score for each data point, enabling flexible trade-offs across objectives.

- **Gradient-space combination:** To address the alignment tax more fundamentally, we can further examine the problem from a gradient perspective. Jointly optimizing multiple objectives often leads to conflicting optimization directions. This observation suggests that treating each objective independently is suboptimal. Instead, one should identify a shared descent direction in gradient space that combines gradients from multiple validation objectives in a principled way.

- **Integration with existing multi-objective optimization methods:** Given the demonstrated effectiveness of our method in the single-objective setting, combining InfOES with existing techniques in multi-objective optimization represents a promising direction. Such integration may yield more stable and interpretable selection strategies under competing alignment goals.

Extension to other domains beyond alignment. Following other papers on alignment and RLHF, we focus on helpfulness and harmlessness in our experiments. However, InfOES is technically generalizable to other domains such as reasoning and chat. For example, one can replace the validation sets representing alignment objectives with sets reflecting math-related skills and extend InfOES to the RLVR paradigm. Interestingly, as shown in our case study, InfOES can already effectively identify math-related data, suggesting that this is a promising direction for future exploration.

We conduct a toy experiment to validate the applicability of InfOES beyond alignment tasks. To assess its potential for enhancing mathematical reasoning capabilities in RLVR, we use a subset of 30k samples from the WebInstruct-verified dataset[2], which covers multiple reasoning domains including mathematics, coding, physics, chemistry, finance, and humanities. We select 8 mathematics-related training samples as the validation set $\mathcal{D}_{\text{val}}$ to represent the objective of improving mathematical performance. Using Qwen2.5-MATH-1.5B as the base model and GRPO as the training backbone, we employ the General-verifier[3] for answer verification. The model is evaluated on the GSM8K test set, with all other hyperparameters following the same settings as in the main RLHF experiments.

Method	Performance (↑)	Wall-clock time (↓)
Standard RL	75.60	811.9
InfOES	77.04	746.9

Table 12: Comparison of performance and wall-clock time between standard RL and InfOES on mathematical reasoning tasks.

As shown in Table 12, InfOES achieves competitive performance compared to full-data training while requiring less wall-clock time, demonstrating its potential applicability for mathematical reasoning tasks.

Extension to full fine-tuning. Following prior work on data influence estimation for LLMs (Xia et al., 2024a), we employ LoRA in our experiments for computational efficiency. However, we believe our method is not inherently restricted to LoRA fine-tuning and can be extended to full fine-tuning. We analyze this extension from both effectiveness and efficiency perspectives.

Effectiveness. Previous work (Xia et al., 2024a) has shown that gradients from full fine-tuning are more informative for data selection than those from LoRA-based instruction tuning. In general, larger gradient dimensions better capture the optimization direction, leading to improved performance. By analogy, our method applied to full fine-tuning would similarly benefit from higher-dimensional gradients.

Efficiency. Our method introduces two main additional costs relative to standard training: (1) computing gradients on the validation set, and (2) influence estimation. Both costs can be substantially reduced in full fine-tuning:

- The validation set does not need to be large. Our experiments show that computing gradients for as few as 8 validation samples is sufficient.
- Gradient dimensionality can be reduced via techniques such as random projection, potentially down to the same scale as LoRA, which greatly reduces the cost of influence computation.

Thus, while we adopt LoRA for practicality, the core methodology remains applicable to full fine-tuning, offering a promising direction for scenarios where computational budgets allow.

[2] https://huggingface.co/datasets/TIGER-Lab/WebInstruct-verified

[3] https://huggingface.co/TIGER-Lab/general-verifier