DUDS: Dual-stage Data Selection for Efficient Reinforcement Learning with Verifiable Rewards

Hongling Zheng[1,2] **Li Shen**[3] **Zichuan Lin**[4] **Jiafei Lyu**[4] **Zhicong Lu**[5]
Shuhan Xu[1] **Yong Luo**[1†] **Deheng Ye**[4] **Dacheng Tao**[2]
[1]School of Computer Science, National Engineering Research Center of Multimedia Software and Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan, China
[2]Nanyang Technological University, Singapore
[3]Shenzhen Campus of Sun Yat-sen University, Shenzhen, China
[4]Tencent Inc., Shenzhen, China [5]University of Chinese Academy of Sciences, Beijing, China
{hlzheng, luoyong, xushuhan}@whu.edu.cn
{mathshenli,lzcthu12, nazaritelzc, ydyl1991, dacheng.tao}@gmail.com

Abstract

Recent advances in large language models (LLMs) have leveraged reinforcement learning with verifiable rewards (RLVR) to enhance reasoning capabilities. However, RLVR typically relies on massive training data and extensive rollouts, posing substantial challenges to computational resources. Existing data selection approaches address this challenge in isolation: offline methods perform static, one-time dataset pruning that cannot adapt to the model's evolving learning needs throughout training, while online methods conduct per-iteration filtering at the expense of significant additional computation. In this paper, we propose DUDS, a **Du**al-stage **D**ata **S**election framework for RLVR, which organically integrates the strengths of both paradigms to improve training efficiency while maintaining competitive performance. Specifically, the offline stage curates a candidate pool from the full dataset by jointly considering quality score and prompt diversity via Determinantal Point Processes, providing an informative warm start for subsequent training. The online stage employs Bayesian posterior updates to estimate prompt pass rates without introducing additional rollouts, and combines them with a freshness metric for data sampling, then applies asymmetric filtering to phase out mastered or intractable problems. Extensive experiments across four reasoning benchmarks demonstrate that DUDS consistently outperforms existing methods in both offline and online data-selection scenarios, achieving competitive performance with significantly less data and computation.

1 Introduction

Recently, reinforcement learning with verifiable rewards (RLVR) [1] has emerged as a prominent technique to unlock the reasoning capabilities of large language models (LLMs). In RLVR training, LLMs explore multiple reasoning trajectories and receive binary rewards based on the correctness of the final answer, iteratively refining their policies. Despite its effectiveness, this paradigm requires a substantial number of rollouts, thereby significantly increasing time and computational costs. Moreover, since not all training samples contribute equally to policy improvement, the mismatch between sub-optimal or redundant data and the model's learning needs further exacerbates data inefficiency. This motivates the design of effective data selection strategies that identify high-quality

[†]Corresponding authors.

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

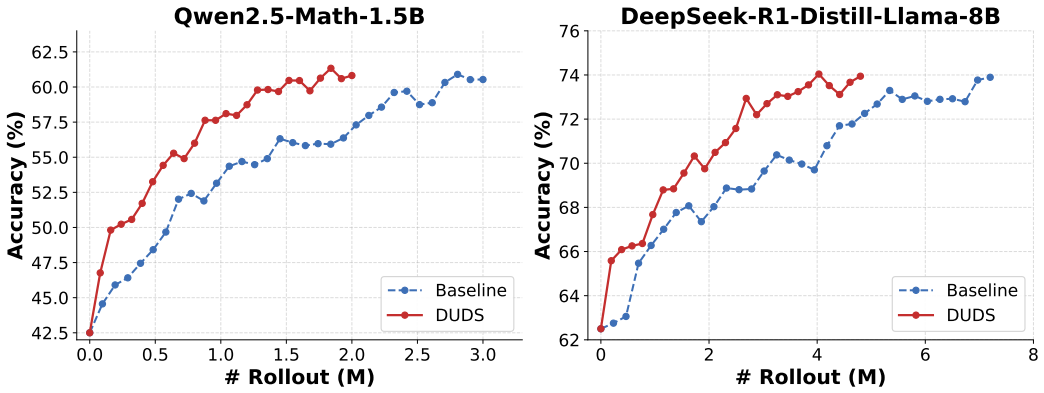

Figure 1: We train Qwen2.5-Math-1.5B and DeepSeek-R1-Distill-Llama-8B on the DAPO-MATH dataset and evaluate them on four math reasoning benchmarks. Compared to the baseline (GRESO), DUDS significantly reduces rollout cost while preserving its performance advantage.

samples tailored to the characteristics of RLVR training, thereby improving training efficiency without sacrificing model performance.

Existing work on data selection in the RLVR setting can be broadly categorized into offline and online methods. Offline methods primarily perform a one-time, static pruning of the dataset through techniques such as rule-based filtering [2] and gradient alignment [3], after which the resulting dataset is used for training. However, RLVR training inherently involves credit assignment and policy evaluation, causing the model's data requirements to evolve dynamically during training. Consequently, the static training data obtained through offline selection can hardly satisfy the model's learning needs across the entire training cycle. Online methods take a step further by incorporating dynamic data selection tailored to RLVR. These approaches perform real-time data filtering before each training iteration through techniques such as prompt oversampling [4, 5], difficulty-referenced evaluation [6], and token entropy estimation [7] to obtain high-quality training data. Since trajectory rollouts and data filtering must be conducted at every iteration, online methods essentially achieve stable performance at the cost of training efficiency. Therefore, a promising direction is to integrate the strengths of both the offline and online paradigms, thereby achieving both training efficiency and strong final performance.

In this paper, we propose a Dual-stage Data Selection (DUDS) framework that integrates the distinct advantages of offline and online phases to serve RLVR. The **offline stage** first leverages the policy model to conduct a holistic evaluation of the entire training dataset, quantifying the model's initial mastery of each problem. We employ the Determinantal Point Process to jointly incorporate a Gaussian quality score derived from pass rates and a semantic similarity matrix constructed from prompt embeddings, naturally balancing appropriate difficulty and diversity. Through this process, we curate an expandable candidate pool from the original dataset, providing an efficient warm-start for the online stage and substantially reducing the burden of data selection. The **online stage** first updates the Bayesian posterior estimate of each prompt's pass rate based on the previous round's rollout results, then performs weighted sampling at each training iteration according to the updated estimates and a freshness metric, ensuring that data selection focuses on samples near the model's current capability boundary while preventing potentially valuable samples from being neglected over prolonged periods. An asymmetric zero-variance filtering mechanism is then applied to the sampled data, phasing out already-mastered samples more aggressively while retaining currently unsolvable ones with greater patience. The main contributions are as follows:

- We revisit the limitations of existing RLVR methods in terms of computational cost and data efficiency, and identify the complementary strengths and limitations of offline and online data selection paradigms.
- We propose DUDS, a dual-stage organically integrated data selection framework, and provide empirical and theoretical analyses to characterize the key designs of each stage, substantially reducing the computational cost of RLVR while maintaining strong model performance.

- Extensive experiments across four reasoning benchmarks demonstrate the effectiveness and efficiency of DUDS under both offline and online data selection scenarios.

2 Related Work

Reinforcement Learning with Verifiable Reward. RLVR has emerged as a promising paradigm for aligning LLMs with desired behaviors and enhancing their complex reasoning capabilities, particularly in domains such as mathematics [8] and code generation [9]. The key advantage of this approach lies in its reward design, which relies solely on simple verification functions to provide binary rewards without requiring learned reward models. By eliminating the need for a costly value network, GRPO [1] offers a more efficient RL framework and demonstrates strong gains in scaling LLMs' reasoning abilities. Building on GRPO, subsequent works have further advanced RLVR by refining various aspects, including loss functions [10, 4], token-level entropy [11], prompt oversampling [5], and hyperparameter tuning [12]. In this work, we focus on improving the data efficiency of RLVR to reduce computational costs while maintaining model performance.

Data Selection for LLMs. Data selection has long been recognized as a critical factor in determining LLMs' performance, directly affecting training efficiency and generalization. A substantial body of work studies data selection in the context of pretraining and supervised fine-tuning (SFT), focusing on identifying informative or high-quality samples [13, 14, 15]. Due to the repeated rollouts and credit assignment involved during training, RLVR incurs significantly greater computational overhead and is more data-sensitive than standard SFT, rendering uniform sampling inadequate for the RLVR setting. Consequently, developing effective data selection methods to identify high-quality prompts in the RLVR scenario has become a key research focus. Early methods mostly rely on heuristic metrics, such as reward trends [16] and reward variance [17], and gradient alignment [18] to perform static filtering of the dataset during an offline stage. These methods fail to account for the model's dynamically evolving data quality requirements throughout training, and often introduce excessive computational overhead while overly pruning the dataset. To address these limitations, recent work explores online data selection strategies that adapt to the current policy. GRESO [5] employs a probabilistic filtering strategy to exclude samples with historically zero variance. Although this approach improves efficiency, it only focuses on historical zero-variance samples and fails to distinguish among the remaining samples. In this work, we enhance the data efficiency of RLVR by integrating optimized strategies for both offline and online data selection.

3 Preliminary

Reinforcement Learning with Verifiable Rewards. Reinforcement Learning with Verifiable Rewards (RLVR) [1] is a training paradigm that enhances the reasoning capabilities of large language models through outcome-based reinforcement learning. Unlike conventional RLHF approaches that rely on learned reward models, RLVR leverages deterministic, rule-based reward functions $r(x, o, y^*) \in \{0, 1\}$ that verify the correctness of a candidate response o against a ground-truth solution y^*. Given a dataset $\mathcal{D} = \{(x_i, y_i^*)\}_{i=1}^N$ of prompt–solution pairs, the policy model π_θ generates a group of G candidate responses $\{o_1, \ldots, o_G\}$ for each prompt x through repeated sampling, commonly referred to as rollouts. Each response receives a binary reward based on answer correctness, and the resulting reward signals are used to update the policy.

Group Relative Policy Optimization. Group Relative Policy Optimization (GRPO) [8] is a variant of Proximal Policy Optimization (PPO) [19] tailored for language model fine-tuning. Instead of computing advantages using a value function, GRPO normalizes reward scores within groups of responses sampled from the same prompt, thereby largely improving training efficiency. GRPO aims to maximize the following objective:

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}_{(x,y^*)\sim\mathcal{D},\ \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{old}}(\cdot|x)}$$
$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i \mid x)}{\pi_{\theta_{old}}(o_i \mid x)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i \mid x)}{\pi_{\theta_{old}}(o_i \mid x)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta \| \pi_{ref}\right)\right), \tag{1}$$

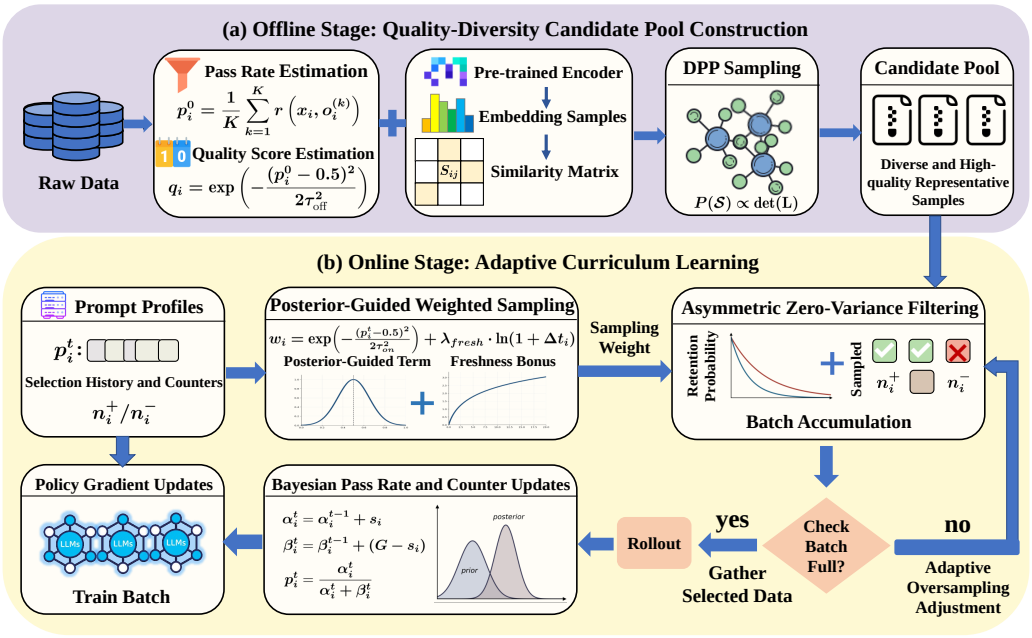

Figure 2: Overview of the DUDS framework. **(a) Offline stage:** We use a Gaussian quality score and prompt diversity to jointly parameterize a DPP kernel for sampling the candidate pool. **(b) Online stage:** at each step, weighted sampling and asymmetric filtering select effective prompts from the pool, and rollout outcomes are then used to update the posterior pass rates via Bayesian inference for the next round.

where A_i is the advantage, computed using the rewards $\{r_1, r_2, \ldots, r_G\}$ corresponding to the G responses within each group:

$$A_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G}) + \delta}, \tag{2}$$

where $r_i = r(x, o_i, y^*)$ is the reward for the i-th response, and $\delta > 0$ is a small constant for numerical stability when all rewards within a group are identical. The advantage of each response is computed as a normalized reward within a group of repeated rollouts.

4 Method

In this section, we present the detailed design of our proposed DUDS framework. In the offline stage, we construct a high-quality, diverse candidate pool based on quality score and prompt diversity, thereby providing a stable starting point for subsequent training. In the online stage, we adopt a dynamic perspective, leveraging posterior estimates of prompt pass rates accumulated from historical rollouts, along with a freshness bonus, to perform weighted sampling and zero-variance filtering over the candidate pool, thereby aligning data selection with the model's evolving training needs.

4.1 Problem Definition

The objective of data selection for RLVR is to identify a high-quality subset from the full training dataset D_{train} for training the LLM policy model π_θ to achieve strong performance on a test dataset D_{test}. Ideally, the selected subset should preserve the model's performance to a level comparable with full-dataset training, while substantially reducing training time and computational cost.

4.2 Offline Stage: Quality-Diversity Candidate Pool Construction

Although the model's data requirements evolve dynamically during training, prior work [17] has demonstrated that a high-quality subset smaller than the full dataset can provide sufficient learning

signals to improve performance. This motivates us to perform a round of lenient data selection in the offline stage, which paves the way for more fine-grained filtering in the subsequent online training.

Quality Score Estimation. The prompt pass rate is commonly adopted as a key indicator of the policy's mastery over a given problem, and it also indirectly reflects the overall quality of the current dataset. For each prompt $x_i \in D_{train}$, the initial policy generates K responses $\{o_i^{(1)}, \ldots, o_i^{(K)}\}$, and the initial pass rate p_i^0 is computed as the average reward over all responses. Numerous studies [6, 20, 21] have shown that prompts with extreme pass rates offer limited utility for the current policy, while those with intermediate pass rates yield the most informative gradient signals for learning. To this end, we define a Gaussian-based quality score q_i to identify prompts whose pass rates lie near the intermediate range:

$$p_i^0 = \frac{1}{K}\sum_{k=1}^{K} r(x_i, o_i^{(k)}), \quad q_i = \exp\left(-\frac{(p_i^0 - 0.5)^2}{2\tau_{\text{off}}^2}\right). \tag{3}$$

While assigning higher quality scores to prompts with intermediate pass rates so as to filter out sub-optimal samples, we set $\tau_{\text{off}} = 0.5$ to gently moderate the scores of prompts at both extremes, avoiding overly strict filtering of prompts with specific pass rates.

Prompt Diversity Characterization. The full dataset inevitably contains a large number of highly similar prompts, which not only increase the rollout burden but also introduce redundant gradient information. A few recent works [22, 23] have attempted to control sample diversity during the online training, but repeatedly computing pairwise similarities within each training batch introduces prohibitive overhead. As prompt diversity is a relatively static property that remains invariant to the model's evolving needs, we directly incorporate it into the offline stage, ensuring that every subsequent sampling step naturally benefits from a well-distributed problem space without requiring additional runtime mechanisms. We employ the contrastive learning-based pre-trained encoder `all-MiniLM-L6-v2` to obtain normalized embeddings $e_i \in R^d$ for each prompt i, based on which we construct a pairwise similarity matrix $S_{ij} = \mathbf{e}_i^\top \mathbf{e}_j$ to characterize prompt diversity.

Quality-Diversity Joint Selection. We employ the Determinantal Point Process (DPP) [24] to jointly incorporate prompt quality and diversity for candidate pool construction. The DPP is parameterized by a positive semi-definite kernel matrix $\mathbf{L}$ that assigns each subset $\mathcal{S} \subseteq \mathcal{D}$ a selection probability proportional to the determinant of its principal submatrix:

$$P(\mathcal{S}) \propto \det(\mathbf{L}_{\mathcal{S}}). \tag{4}$$

We construct the kernel via the standard quality–diversity decomposition $L_{ij} = q_i \cdot S_{ij} \cdot q_j$. The diagonal entry $L_{ii} = q_i^2$ governs the marginal inclusion probability of each sample, thereby reflecting its quality, while the off-diagonal entry L_{ij} encodes pairwise redundancy. The determinant simultaneously rewards high diagonal values and penalizes large off-diagonal correlations, achieving the desired quality–diversity trade-off without any explicit balancing hyperparameter. Since exact inference for DPP is NP-hard, we resort to a greedy algorithm: starting from the empty set, we iteratively select the element that yields the largest marginal gain in the log-determinant until the target size is reached, yielding the final candidate pool.

It is worth noting that, for mathematical reasoning tasks, problems with highly similar semantic expressions may yield substantially different pass rates due to variations in their underlying difficulty or required reasoning skills. Such an observation precisely motivates our joint incorporation of the quality score and prompt diversity, where the quality score provides an initial discrimination of samples from the perspective of the model's capability boundary, while the diversity metric further eliminates redundant samples within the same difficulty range. The complementarity of these two metrics ensures that the final candidate pool achieves balanced coverage along both the quality and semantic dimensions.

4.3 Online Stage: Adaptive Curriculum Learning

During the online training phase, we maintain a profile for each prompt x_i that contains its estimated pass rate $p_i^{(t)}$, selection history, and consecutive zero-variance counts, which are updated based on the rollout results from the previous round. The model then performs weighted sampling of prompts at each iteration and conducts rollouts after filtering out zero-variance prompts that fail to produce effective training signals. Built on the high-quality candidate pool, the online phase

achieves significant improvements in both training efficiency and model performance by integrating lightweight and fine-grained data selection steps.

Posterior-Guided Weighted Sampling. Considering the significant computational cost introduced by estimating sample quality in real time at each round, instead of using additional rollouts, we employ the Bayesian posterior estimate of each prompt's pass rate [25] as the core feature. The estimated pass rate is modeled as $p_i^t \sim \text{Beta}(\alpha_i^t, \beta_i^t)$, defined as the estimate held before the step-t rollout begins. At $t = 1$, the distribution is set to the prior derived from the empirical pass rate p_i^0: $\alpha_i^1 = \kappa \cdot p_i^0 + 1$ and $\beta_i^1 = \kappa \cdot (1 - p_i^0) + 1$, where κ controls the prior strength and the additive constant ensures a proper distribution when $p_i^0 \in \{0, 1\}$. For $t \geq 2$, let s_i denote the number of successes out of G trials observed during the step-$(t-1)$ rollout. We incorporate this outcome into the posterior to obtain the estimate entering step t:

$$\alpha_i^t = \alpha_i^{t-1} + s_i, \quad \beta_i^t = \beta_i^{t-1} + (G - s_i), \quad p_i^t = \frac{\alpha_i^t}{\alpha_i^t + \beta_i^t}, \tag{5}$$

Through this update, p_i^t aggregates the offline prior together with all past rollout observations. Unlike exponential moving averages with a fixed decay, the Bayesian posterior has variance scaling as $1/(\alpha_i^t + \beta_i^t)$, anchoring estimates to the prior under scarce evidence while becoming data-dominated as observations accumulate. To further align with the training characteristics of the online phase, we introduce a freshness metric to ensure that potentially important samples are not neglected over extended periods. The posterior estimate and freshness metric are then combined through weighted aggregation to obtain the sampling weight:

$$w_i = \exp\left(-\frac{(p_i^t - 0.5)^2}{2\tau_{on}^2}\right) + \lambda_{fresh} \cdot \ln(1 + \Delta t_i), \tag{6}$$

where $\Delta t_i = t - t_i^{\text{last}}$ denotes the number of steps since prompt x_i was last selected, and λ_{fresh} controls the magnitude of the recency bonus. Notably, the first term differs from the quality score used in the offline phase in that we further narrow the Gaussian bandwidth τ_{on} to select samples with the richest learning signals. We theoretically analyze the reasonable range of τ_{on} in Appendix A and empirically verify it in our experiments.

Asymmetric Zero-Variance Filtering. While the weighted sampling strategy prioritizes prompts with moderate pass rates to exclude sub-optimal data, some sampled prompts may still yield all-correct or all-incorrect rollouts, providing no gradient signal while occupying valuable computational resources. To address this, after sampling, we further filter prompts based on their history of zero-variance outcomes. Specifically, we maintain two consecutive zero-variance counters n_i^+ and n_i^- for each prompt x_i, recording the number of consecutive all-correct and all-incorrect outcomes, respectively. The retention probability is then defined as:

$$P(\text{retain} \mid x_i) = \begin{cases} \gamma_+^{n_i^+}, & \text{if last } n_i^+ \text{ outcomes are all-correct,} \\ \gamma_-^{n_i^-}, & \text{if last } n_i^- \text{ outcomes are all-incorrect,} \\ 1, & \text{otherwise,} \end{cases} \tag{7}$$

Since problems that the model is temporarily unable to solve typically contain richer information and thus deserve more learning patience than those it can already handle well, in practice we set γ_- slightly larger than γ_+. If a single sampling-filtering cycle does not yield enough effective samples to fill the training batch, additional cycles are executed, with the oversampling ratio ρ adaptively adjusted based on the observed filtering rate: $\rho^{(t+1)} = \max(1.0,\ 1/(1 - \hat{f}^{(t)}))$, where $\hat{f}^{(t)}$ is the empirical fraction of discarded prompts.

5 Experiment

Models & Baselines. We conduct experiments on three models: Qwen2.5-Math-1.5B [26], Qwen2.5-Math-7B [26], and DeepSeek-R1-Distill-Llama-8B [1]. We select several representative offline and online data selection methods as baselines, including offline SFT methods PPL-Top [27] and PPL-Middle [28], offline RLVR methods LIMR [3] and Learnalign [18], online random selection method Random, online RLVR methods Efficient-RL [6] and GRESO [5]. In addition, "Base" denotes the

test performance of the initial model without any RL training, and "Full" denotes the test performance of the model trained on the entire training set without data selection. To ensure a fair comparison, the online baseline methods are applied to the same subset selected by DUDS during the offline stage.

Training & Evaluation. We train the model on the DAPO-Math dataset [4] using the GRPO algorithm. To evaluate the performance of the trained models, we use four widely adopted mathematical reasoning benchmarks: Math500 [29], AIME25 [30], AMC [31], and Olympiad Bench [32], and report the performance of the checkpoint that achieves the best average score across all benchmarks. All experiments are carried out on a server with 32 NVIDIA H20 GPUs, each with 96GB of memory. The experimental hyperparameter configurations of DUDS are shown in Table 5.

Table 1: Performance comparison of various data selection methods. "Ratio" and "RN" denote the ratio of selected data and online rollout numbers, respectively.

Model		**Method**	**Accuracy**					**Efficiency**	
			Math500	**AIME25**	**AMC**	**Olympiad**	**Average**	**Ratio**	**RN**
Qwen2.5 -Math-1.5B		Base	54.6	8.4	42.5	29.1	33.7	-	-
		Full	75.8	15.3	62.2	39.6	48.2	100%	100%
	Offline	PPL-Top	65.8	10.8	50.1	30.8	39.4	20%	100%
		PPL-Middle	69.5	11.2	54.0	32.8	41.9	20%	100%
		LIMR	**73.5**	12.3	58.2	35.0	44.8	20%	100%
		Learnalign	70.1	13.8	59.9	34.8	44.7	20%	100%
		DUDS	73.2	**15.0**	**61.8**	**38.9**	**47.2**	20%	100%
	Online	Random	72.5	11.8	56.2	34.5	43.8	20%	60%
		Efficient-RL	73.2	13.0	59.5	36.2	45.5	20%	60%
		GRESO	73.6	15.0	61.4	38.5	47.1	20%	60%
		DUDS	**74.0**	**15.3**	**62.0**	**38.8**	**47.5**	20%	40%
Qwen2.5 -Math-7B		Base	64.5	18.4	52.5	33.5	42.2	-	-
		Full	86.8	24.3	68.9	47.0	56.8	100%	100%
	Offline	PPL-Top	76.2	13.8	55.8	34.8	45.2	20%	100%
		PPL-Middle	76.0	14.3	57.2	37.0	46.1	20%	100%
		LIMR	78.0	15.8	63.8	39.8	49.4	20%	100%
		Learnalign	80.5	17.2	**64.1**	39.8	50.4	20%	100%
		DUDS	**82.8**	**17.4**	60.2	**41.5**	**50.5**	20%	100%
	Online	Random	74.3	14.5	56.3	38.5	45.9	20%	60%
		Efficient-RL	79.8	15.2	59.5	41.0	48.9	20%	60%
		GRESO	82.2	16.8	60.7	44.1	51.0	20%	60%
		DUDS	**82.5**	**17.3**	**61.1**	**45.6**	**51.6**	20%	40%
DeepSeek-R1 -Distill -Llama-8B		Base	88.5	30.4	62.5	37.2	54.7	-	-
		Full	94.8	43.1	75.8	48.5	65.6	100%	100%
	Offline	PPL-Top	89.4	36.3	66.3	38.5	57.6	20%	100%
		PPL-Middle	90.1	37.2	67.0	40.6	58.7	20%	100%
		LIMR	90.6	39.9	70.5	41.2	60.6	20%	100%
		Learnalign	91.0	**41.8**	70.2	42.8	61.5	20%	100%
		DUDS	**92.2**	40.5	**71.0**	**42.8**	**61.6**	20%	100%
	Online	Random	90.1	40.4	67.8	40.5	59.7	20%	60%
		Efficient-RL	91.6	40.6	73.6	42.1	62.0	20%	60%
		GRESO	92.0	42.2	73.5	**43.3**	62.8	20%	60%
		DUDS	**92.1**	**42.5**	**73.9**	43.1	**62.9**	20%	40%

Main Results. As shown in Table 1, we evaluate DUDS under both offline and online settings. In the offline setting, models are trained and evaluated with standard GRPO directly on the selected offline subset, without introducing any further online-stage data selection. Although SFT-based selection methods leverage additional information from training data, their performance remains relatively poor due to the mismatch between SFT and RL objectives. In contrast, DUDS organically integrates prompt quality and diversity into the construction of the offline candidate pool, achieving superior performance over all baselines using only 20% of the original data, and remaining competitive with full-dataset training. This demonstrates that the offline stage of DUDS effectively preserves the most

informative training signals, providing a strong starting point for reducing computational cost in the online stage. In the online training stage, DUDS further refines data selection through weighted sampling and asymmetric zero-variance filtering, reducing the rollout cost to 40% of the original budget while consistently outperforming other online selection methods across most model–task combinations. Notably, in a few cases the online variant of DUDS performs marginally worse than its offline counterpart. We attribute this to the high quality of the offline candidate pool, which already approaches a near-optimal upper bound; under such conditions, the marginal benefit of finer online sampling is insufficient to offset the information loss induced by the reduced rollout budget.

Table 2: Component ablation on Qwen2.5-Math-1.5B. The three components are DPP quality-diversity selection, posterior-guided sampling, and asymmetric zero-variance filtering. Online RN denotes the online rollout count relative to the Full baseline. The first row provides a full-budget reference; all remaining rows use a 40% budget.

Quality-diversity	Posterior-guided	Asymmetric filtering	AIME25	AMC	Olympiad	Online RN
✓			15.0	61.8	38.9	100%
✓			11.2	55.4	33.7	40%
✓	✓		15.1	61.5	38.6	40%
✓		✓	14.8	61.2	38.4	40%
✓	✓	✓	**15.3**	**62.0**	**38.8**	40%

Role of Different Components. As shown in Table 2, we examine the contributions of the online components built upon the DPP-selected candidate pool. The first row provides a full-budget reference using conventional GRPO training without online selection. Reducing the rollout budget to 40% without either online component leads to a substantial performance drop, as shown in the second row. Under the same reduced budget, adding either posterior-guided sampling or asymmetric filtering substantially improves performance, while combining both components achieves the best results across all three benchmarks among the 40%-budget variants, as shown in the fifth row. The complete framework also achieves performance comparable to the full-budget reference, demonstrating that the two online components jointly reduce the rollout requirements while preserving model performance.

Computational Cost Analysis. Table 3 compares the online training time breakdown of GRESO and DUDS across three different stages, including model update, rollout generation, and other overheads. Across all three backbone models, DUDS significantly reduces rollout time while incurring only a marginal increase in other overheads, resulting in a substantial reduction in total training time. Although DUDS introduces additional components that slightly increase deployment overhead, this cost is fully acceptable given the significant improvement in computational efficiency. These results demonstrate that DUDS achieves a favorable trade-off between computational efficiency and training effectiveness, confirming that our design effectively reduces the rollout burden without sacrificing model performance. We further report the complete end-to-end cost, including offline data selection, in Table 8, showing that DUDS retains its efficiency advantage after accounting for preprocessing overhead.

Table 3: Computational cost analysis in the online training stage (hours).

Method	Training	Other	Rollout	Total
Qwen2.5-Math-1.5B				
GRESO	12.5	2.8	19.4	34.7
DUDS	12.0	3.4	12.9	**28.3**
Qwen2.5-Math-7B				
GRESO	33.6	7.0	53.2	93.8
DUDS	32.5	8.2	35.5	**76.2**
DeepSeek-R1-Distill-Llama-8B				
GRESO	36.8	7.6	58.4	102.8
DUDS	35.6	8.9	38.9	**83.4**

Effect of Offline Candidate Pool Size. The offline candidate pool size $|\mathcal{C}|$ governs the trade-off between computational cost and data coverage in the online training stage. While a larger candidate pool provides richer gradient information, it inevitably increases the burden of profile maintenance. We vary $|\mathcal{C}|$ over $\{10\%, 20\%, 50\%, 100\%\}$ with all online-stage hyperparameters fixed. It is worth noting that the primary objective of the DUDS framework is to reduce the number of rollouts and the overall computational overhead, where the offline stage serves the online stage. Therefore, we

determine the hyperparameters mainly according to the model's performance in the online stage. As shown in Figure 3, we observe that the model achieves favorable performance when the candidate pool size is around 20%. Beyond this point, further increasing $|\mathcal{C}|$ leads to a noticeable rise in computational cost without yielding clear performance gains. Accordingly, we adopt $|\mathcal{C}| = 20\%$ as the default setting, which provides the best balance between performance and efficiency.

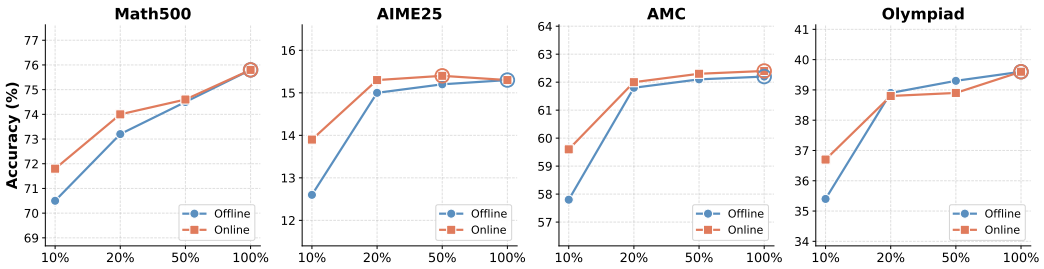

Figure 3: Ablation study on the candidate pool size based on Qwen2.5-Math-1.5B. "Offline" denotes RL training directly on the candidate pool without online selection, while "Online" further applies the full DUDS pipeline.

Effect of Gaussian Bandwidth τ_{on}. The online Gaussian bandwidth τ_{on} controls how sharply DUDS concentrates sampling around prompts whose estimated pass rate is close to 0.5. As shown in Table 6, we sweep $\tau_{\text{on}} \in \{0.05, 0.1, 0.15, 0.2, 0.3, 0.5\}$ and observe that the performance follows a clear non-monotonic trend, where overly small bandwidths overfit to a narrow band of prompts and neglect harder examples, while overly large bandwidths dilute the gradient signal by approaching uniform sampling. The best performance is achieved around $\tau_{\text{on}} = 0.1$ to 0.15, which closely matches the theoretical derivation provided in Appendix A. Accordingly, we adopt $\tau_{\text{on}} = 0.1$ as the default value across all experiments.

Robustness across Diverse RLVR Training Datasets. To evaluate the effect of different training datasets, we conduct experiments using three commonly used RLVR datasets, including HARP [33] (5k samples), Open-R1 [34] (30k samples), and DeepScaleR [35] (40k samples). As illustrated in Figure 4, performance improves consistently as the size of the training dataset increases, indicating that training on larger high-quality corpora can effectively enhance reasoning capabilities. More importantly, DUDS achieves strong performance across all three datasets, consistently outperforming competitive baselines in both offline and online scenarios. These results confirm that DUDS generalizes well across training data of different sources and scales, demonstrating its robustness and broad applicability for improving data efficiency in RLVR training.

Scalability Study. To verify the scalability of DUDS, we further evaluate it on larger backbone models, including Qwen2.5-14B and Qwen2.5-32B. The reported accuracy denotes the average performance across the four benchmarks listed in Table 1. As shown in Figure 5, DUDS converges to a comparable or slightly higher accuracy than GRESO on both models, while consuming significantly fewer rollouts. Such consistent advantages across both 14B and 32B settings indicate that the benefits of DUDS do not diminish with model size but rather generalize well to larger architectures. These results confirm that DUDS maintains its efficiency advantage at scale, providing a reliable solution for reducing the rollout burden of RLVR training on increasingly larger LLMs.

Table 4: Ablation of prompt diversity and freshness on Qwen2.5-Math-1.5B. All variants use a 20% candidate pool and a 40% online rollout budget. Removing prompt diversity replaces DPP selection with top-20% selection by quality score; removing freshness sets $\lambda_{\text{fresh}} = 0$.

Variant	Math500	AIME25	AMC	Olympiad	Average
w/o prompt diversity	73.0	14.2	60.8	37.5	46.4
w/o freshness	73.5	14.8	61.3	38.1	46.9
w/o both	72.6	13.6	59.8	36.9	45.7
DUDS (full)	**74.0**	**15.3**	**62.0**	**38.8**	**47.5**

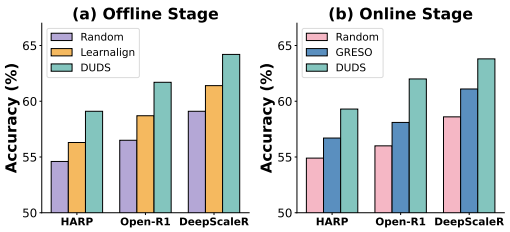

Figure 4: Robustness of DUDS across diverse RLVR training datasets on Qwen2.5-Math-7B. Accuracy (%) is averaged over the four reasoning benchmarks. (a) Offline data selection scenario. (b) Online data selection scenario.

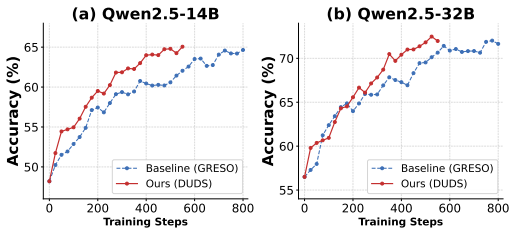

Figure 5: Scaling to 14B and 32B models. DUDS and GRESO exhibit similar convergence speed, while DUDS reaches comparable accuracy with substantially fewer rollouts.

Effect of Prompt Diversity and Freshness. To distinguish the contributions of prompt diversity and freshness from pass-rate-based selection, we ablate these two components while keeping the remaining settings unchanged. As shown in Table 4, removing prompt diversity reduces average accuracy from 47.5 to 46.4, indicating that diversity-aware candidate pool construction provides benefits beyond quality-based selection alone. Removing the freshness bonus decreases average accuracy to 46.9, supporting the utility of encouraging the revisiting of prompts that have not been selected recently. Removing both components further reduces average accuracy to 45.7, with performance declining across all four benchmarks. These results show that prompt diversity and freshness complement the underlying pass-rate-based selection strategy in improving DUDS.

6 Conclusion

In this paper, we propose DUDS, a dual-stage data selection framework for reinforcement learning with verifiable rewards. The offline stage leverages Determinantal Point Processes to jointly incorporate a Gaussian quality score and semantic similarity, thereby curating a diverse, quality-aware candidate pool as an efficient warm start. The online stage dynamically adapts to the model's evolving capabilities via Bayesian posterior pass rate estimation, posterior-guided weighted sampling with a freshness bonus, and asymmetric zero-variance filtering. Extensive experiments across four reasoning benchmarks demonstrate that DUDS consistently achieves competitive or superior performance with significantly less data and computational resources than existing methods.

Limitations. The initial rollout for pass rate estimation introduces a one-time computational overhead. Additionally, our framework currently focuses on mathematical reasoning with binary rewards. Extending it to domains with softer reward signals remains a promising future direction.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (Grant No. U23A20318 and 62276195) and the Foundation for Innovative Research Groups of Hubei Province (Grant No. 2024AFA017). Li Shen is supported by NSFC Grant (No. 62576364), GuangDong Basic and Applied Basic Research Foundation (2026B1515020071), CCF-Tencent Rhino-Bird Open Research Fund (NO. CCF-Tencent RAGR20250114) and Tencent JR2025TEG002, CAAI-Tencent Rhino-Bird Open Research Fund (NO. 2026CAAI-Tencent-08). This work was supported by WHU-Kingsoft Joint Lab. The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

References

[1] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[2] Baizhou Huang and Xiaojun Wan. PROS: Towards compute-efficient RLVR via rollout prefix reuse. In *The Fourteenth International Conference on Learning Representations*, 2026.

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

[4] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[5] Haizhong Zheng, Yang Zhou, Brian R. Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for LLM reasoning via selective rollouts. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[6] Yifan Sun, Jingyan Shen, Yibin Wang, Tianyu Chen, Zhendong Wang, Mingyuan Zhou, and Huan Zhang. Improving data efficiency for LLM reinforcement fine-tuning through difficulty-targeted online data selection and rollout replay. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[7] Haoming Meng, Kexin Huang, Shaohang Wei, Chiyu Ma, Shuo Yang, Xue Wang, Guoyin Wang, Bolin Ding, and Jingren Zhou. Sparse but critical: A token-level analysis of distributional shifts in RLVR fine-tuning of LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026.

[8] 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.

[9] Yaoyu Zhu, Di Huang, Hanqi Lyu, Xiaoyun Zhang, Chongxiao Li, Wenxuan Shi, Yutong Wu, Jianan Mu, Jinghua Wang, Yang zhao, Pengwei Jin, Shuyao Cheng, shengwen Liang, Xishan Zhang, Rui Zhang, Zidong Du, Qi Guo, Xing Hu, and Yunji Chen. Qimeng-codev-r1: Reasoning-enhanced verilog generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[10] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025.

[11] Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[12] Zihe Liu, Jiashun Liu, Yancheng He, Weixun Wang, Jiaheng Liu, Ling Pan, Xinyu Hu, Shaopan Xiong, Ju Huang, Jian Hu, Shengyi Huang, Siran Yang, Jiamang Wang, Wenbo Su, and Bo Zheng. Tricks or traps? a deep dive into RL for LLM reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026.

[13] Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. 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 7602–7635, 2024.

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

[15] Shaobo Wang, Xiangqi Jin, Ziming Wang, Jize Wang, Jiajun Zhang, Kaixin Li, Zichen Wen, Zhong Li, Conghui He, Xuming Hu, et al. Data whisperer: Efficient data selection for task-specific llm fine-tuning via few-shot in-context learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23287–23305, 2025.

[16] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025.

[17] Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, Weizhu Chen, Shuohang Wang, Simon Shaolei Du, and yelong shen. Reinforcement learning for reasoning in large language models with one training example. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[18] Shipeng Li, Shikun Li, Zhiqin Yang, Xinghua Zhang, Gaode Chen, Xiaobo Xia, Hengyu Liu, and Zhe Peng. Learnalign: Reasoning data selection for reinforcement learning in large language models based on improved gradient alignment. *arXiv preprint arXiv:2506.11480*, 2025.

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

[20] Hao Yi, Yulan Hu, Xin Li, Sheng Ouyang, Lizhong Ding, and Yong Liu. Learn more with less: Uncertainty consistency guided query selection for RLVR. In *The Fourteenth International Conference on Learning Representations*, 2026.

[21] Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, JeongYeon Nam, and Donghyun Kwak. Online difficulty filtering for reasoning oriented reinforcement learning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 700–719, 2026.

[22] Ziyu Ye, Rishabh Agarwal, Tianqi Liu, Rishabh Joshi, Sarmishta Velury, Quoc V Le, Qijun Tan, and Yuan Liu. Reward-guided prompt evolving in reinforcement learning for llms. In *Forty-second International Conference on Machine Learning*, 2025.

[23] Jian Yao, Ran Cheng, Xingyu Wu, Jibin Wu, and KC Tan. Diversity-aware policy optimization for large language model reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[24] Raja Hafiz Affandi, Emily Fox, Ryan Adams, and Ben Taskar. Learning the parameters of determinantal point process kernels. In *International Conference on Machine Learning*, pages 1224–1232. PMLR, 2014.

[25] Rens Van de Schoot, Sarah Depaoli, Ruth King, Bianca Kramer, Kaspar Märtens, Mahlet G Tadesse, Marina Vannucci, Andrew Gelman, Duco Veen, Joukje Willemsen, et al. Bayesian statistics and modelling. *Nature Reviews Methods Primers*, 1(1):1, 2021.

[26] An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.

[27] Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, et al. The bigscience roots corpus: A 1.6 tb composite multilingual dataset. *Advances in Neural Information Processing Systems*, 35:31809–31826, 2022.

[28] Zachary Ankner, Cody Blakeney, Kartik Sreenivasan, Max Marion, Matthew L Leavitt, and Mansheej Paul. Perplexed by perplexity: Perplexity-based pruning with small reference models. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024.

[29] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[30] Parshin Shojaee, Seyed Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[31] Paul Zeitz. *The art and craft of problem solving*. John Wiley & Sons, 2016.

[32] Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850, 2024.

[33] Albert S Yue, Lovish Madaan, Ted Moskovitz, DJ Strouse, and Aaditya K Singh. Harp: A challenging human-annotated math reasoning benchmark. *arXiv preprint arXiv:2412.08819*, 2024.

[34] Hugging Face. Open r1: A fully open reproduction of deepseek-r1, january 2025. *URL https://github. com/huggingface/open-r1*, 7, 2025.

[35] Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, et al. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*, 3(5), 2025.

Appendix

The appendix is organized into several sections, each providing additional insights and details related to different aspects of the main work.

A Theoretical Analysis: Optimality of Posterior-Guided Weighted Sampling **14**

A.1 Main Result 14

A.2 Connection to the Gaussian Sampling Weight 15

B Implementation Details and Hyperparameters **15**

C Algorithm Pseudocode **16**

D More Experiments **16**

E Broader Impact **19**

A Theoretical Analysis: Optimality of Posterior-Guided Weighted Sampling

A.1 Main Result

Theorem 1 (Gradient Signal Magnitude under GRPO)**.** *Let $p_i \in (0,1)$ denote the pass rate of prompt x_i, and assume the score function $\nabla_\theta \log \pi_\theta(o \mid x_i)$ has bounded second moment $\mathbb{E}[\|\nabla_\theta \log \pi_\theta\|^2] \leq M$ uniformly across rollouts. Then the expected squared gradient magnitude contributed by prompt x_i satisfies:*

$$\mathbb{E}\left[\|g_i(\theta)\|^2\right] \;\leq\; \frac{M(G-1)}{G^2} \cdot \phi_G(p_i), \tag{8}$$

where the probability of producing a non-zero gradient is exactly:

$$\phi_G(p_i) \;\triangleq\; \Pr[\sigma_i > 0] \;=\; 1 - p_i^G - (1-p_i)^G. \tag{9}$$

The function $\phi_G(p_i)$ is strictly concave on $[0,1]$, symmetric about $p_i = 0.5$, and attains its unique maximum at $p_i^* = 0.5$.

Proof. We decompose the analysis into two parts: (i) the probability of producing a non-zero gradient, and (ii) the conditional gradient magnitude.

Part (i): Probability of non-zero variance. The advantage $A_i^{(g)}$ is well-defined only when $\sigma_i > 0$, i.e., when not all rollouts share identical rewards. The number of correct responses $S_i = \sum_g r_i^{(g)}$ follows a Binomial distribution: $S_i \sim \text{Bin}(G, p_i)$. Zero variance occurs iff $S_i \in \{0, G\}$, so:

$$\Pr[\sigma_i = 0] = \Pr[S_i = 0] + \Pr[S_i = G] = (1-p_i)^G + p_i^G. \tag{10}$$

Hence $\phi_G(p_i) = 1 - p_i^G - (1-p_i)^G$.

To establish concavity, compute the second derivative:

$$\phi_G''(p_i) = -G(G-1)\left[p_i^{G-2} + (1-p_i)^{G-2}\right] < 0, \quad \forall p_i \in (0,1),\, G \geq 2. \tag{11}$$

By symmetry $\phi_G(p_i) = \phi_G(1-p_i)$ and concavity, the unique maximum is at $p_i^* = 0.5$.

Part (ii): Conditional gradient magnitude. Conditional on $\sigma_i > 0$, the normalized advantages $A_i^{(g)}$ satisfy $\sum_g A_i^{(g)} = 0$ and $\sum_g (A_i^{(g)})^2 = G$. By the Cauchy–Schwarz inequality:

$$\|g_i\|^2 = \left\| \frac{1}{G} \sum_g A_i^{(g)} \nabla_\theta \log \pi_\theta(o_i^{(g)}) \right\|^2 \leq \frac{1}{G^2} \sum_g (A_i^{(g)})^2 \cdot \sum_g \|\nabla_\theta \log \pi_\theta(o_i^{(g)})\|^2. \tag{12}$$

Taking expectations and using the bounded score function assumption yields $\mathbb{E}[\|g_i\|^2 \mid \sigma_i > 0] \leq M(G-1)/G$, and combining with the marginal probability $\phi_G(p_i)$ produces the bound in Equation (8). □

A.2 Connection to the Gaussian Sampling Weight

Theorem 1 establishes that the gradient signal peaks at $p_i = 0.5$, motivating a sampling profile concentrated at this point. Two design choices remain: the functional form (we use a Gaussian kernel) and the bandwidth τ_{on}.

Functional form. Among unimodal kernels symmetric about $p_i = 0.5$, the Gaussian is the maximum-entropy distribution with prescribed mean and variance, making it a parsimonious default choice that imposes minimal structural assumption beyond the location and scale of the peak. Importantly, in our online stage, the sampling weight is computed using the *posterior estimate* $\hat{p}_i^{(t)}$ rather than the unobserved true pass rate $p_i^{(t)}$. We argue below that this estimation step, rather than the intrinsic shape of ϕ_G, is the dominant factor governing the optimal bandwidth.

Bandwidth selection: matching posterior uncertainty. Let $\hat{p}_i^{(t)}$ denote the Bayesian posterior mean of the pass rate after Equation (5), modeled as $\hat{p}_i^{(t)} \mid p_i^{(t)} \sim \mathcal{N}(p_i^{(t)}, \sigma_{\hat{p}}^2)$ via a Gaussian approximation to the Beta posterior, with

$$\sigma_{\hat{p}}^2 = \frac{\alpha_i^{(t)} \beta_i^{(t)}}{(\alpha_i^{(t)} + \beta_i^{(t)})^2 \, (\alpha_i^{(t)} + \beta_i^{(t)} + 1)}. \tag{13}$$

The sampling weight $w_i = \exp(-(\hat{p}_i^{(t)} - 0.5)^2/(2\tau^2))$ acts on the noisy estimate, not the true pass rate. The following lemma characterizes the resulting optimal bandwidth.

Lemma 1 (Bandwidth–Uncertainty Matching)**.** *Suppose $p_i^{(t)}$ is locally distributed near 0.5 with effective variance τ_ϕ^2 induced by ϕ_G, and the posterior estimate satisfies $\hat{p}_i^{(t)} \mid p_i^{(t)} \sim \mathcal{N}(p_i^{(t)}, \sigma_{\hat{p}}^2)$. Then the bandwidth τ^* that maximizes the expected gradient signal $\mathbb{E}_{\hat{p}_i^{(t)}}[\phi_G(p_i^{(t)}) \cdot w_i(\hat{p}_i^{(t)}; \tau)]$ satisfies*

$$(\tau^*)^2 \;=\; \tau_\phi^2 + \sigma_{\hat{p}}^2, \tag{14}$$

i.e., the optimal bandwidth jointly accounts for the intrinsic shape of ϕ_G and the posterior estimation uncertainty.

Proof Sketch. The argument follows from the convolution property of Gaussian distributions. Approximating ϕ_G locally near $p = 0.5$ as a Gaussian with variance τ_ϕ^2 and noting that $\hat{p}_i^{(t)}$ is a Gaussian-perturbed observation of $p_i^{(t)}$, the expected weighted signal becomes a Gaussian convolution whose variance is the sum of the two component variances, yielding Equation (14). □

Numerical evaluation. With our default rollout group size $G = 8$ and prior strength $\kappa = G = 8$, the initial Beta parameters at a moderate prior pass rate $p_i^0 = 0.5$ are $\alpha_i^{(1)} = \beta_i^{(1)} = 5$, giving $\sigma_{\hat{p}} \approx 0.151$ at training onset. After several rollout updates ($\alpha + \beta \in [18, 26]$), the posterior standard deviation contracts to the range $\sigma_{\hat{p}} \in [0.097, 0.115]$. The intrinsic shape contribution τ_ϕ from ϕ_G is comparatively minor in this range, so the optimal bandwidth in Equation (14) is dominated by posterior uncertainty:

$$\tau^* \;\approx\; \sigma_{\hat{p}} \;\approx\; 0.10. \tag{15}$$

This theoretical optimum closely matches our empirically chosen $\tau_{\text{on}} = 0.1$, providing a principled justification that explicitly couples the sampling bandwidth to the posterior estimation precision in our Bayesian update scheme.

B Implementation Details and Hyperparameters

For reproducibility, Table 5 lists all hyperparameters used in our experiments. Unless otherwise stated, the same values are applied across all three models (Qwen2.5-Math-1.5B, Qwen2.5-Math-7B, and DeepSeek-R1-Distill-Llama-8B).

Table 5: Hyperparameters used in our experiments.

Category	Hyperparameter	Value
RL Training	RL algorithm	GRPO
	Learning rate	1e-6
	Weight decay	0.01
	Warmup ratio	0.03
	Global batch size (prompts/step) B	1024
	Rollouts per prompt G	8
	Max prompt length	1024
	Max response length	4096
	Clip range ϵ	0.2
	Total training steps T	800
Hardware	Qwen2.5-Math-1.5B GPUs	$8\times$ H20
	Qwen2.5-Math-7B GPUs	$16\times$ H20
	DeepSeek-R1-Distill-Llama-8B GPUs	$16\times$ H20
Offline Stage	Offline rollouts per prompt K	8
	Offline Gaussian bandwidth τ_{off}	0.5
	Candidate pool ratio $\lvert\mathcal{D}_{\text{pool}}\rvert/\lvert\mathcal{D}_{\text{train}}\rvert$	20%
Online Stage	Online Gaussian bandwidth τ_{on}	0.1
	Freshness coefficient λ_{fresh}	0.05
	Bayesian prior strength κ	8
	All-correct decay γ_+	0.3
	All-incorrect decay γ_-	0.5
	Initial oversampling ratio ρ_0	1.5
	Max oversampling cycles $C_{\max}$	3

C Algorithm Pseudocode

D More Experiments

Table 6: Effect of Gaussian bandwidth τ_{on} on Qwen2.5-Math-1.5B.

τ_{on}	Math500	AIME25	AMC	Olympiad	Average
0.05	74.0	13.2	58.8	36.5	45.6
0.10	74.0	**15.3**	**62.0**	**38.8**	**47.5**
0.15	**74.5**	15.1	61.6	38.4	47.4
0.20	73.8	14.5	60.7	37.6	46.7
0.30	73.2	13.5	59.2	36.3	45.6
0.50	72.6	12.4	57.5	34.9	44.4

Generalization to Other 1.5B Backbones. To further verify that the effectiveness of DUDS does not rely on a specific model family, we conduct additional experiments on DeepSeek-R1-Distill-Qwen-1.5B, a distilled reasoning model with a different pretraining recipe from the Qwen2.5-Math series used in our main experiments. As shown in Table 7, DUDS consistently outperforms all offline and online baselines across the four reasoning benchmarks. In the offline setting, DUDS surpasses the strongest baseline (Learnalign) by 1.0 points on average using only 20% of the training data, approaching the performance of full-dataset training. In the online setting, DUDS

Algorithm 1 DUDS: Dual-stage Data Selection for RLVR

Require: Dataset $\mathcal{D}_{\text{train}}$; initial policy π_{θ_0}; group size G; batch size B; pool size N_{pool}; max cycles $C_{\max}$; hyperparameters $\{\tau_{\text{off}}, \tau_{\text{on}}, \kappa, \lambda_{\text{fresh}}, \gamma_+, \gamma_-, \rho_0\}$

// Offline Stage

Compute initial pass rate p_i^0, quality q_i, and embedding $\mathbf{e}_i$ for each x_i

$\mathcal{D}_{\text{pool}} \leftarrow \textsc{GreedyDPP}\big(L_{ij} = q_i(\mathbf{e}_i^\top \mathbf{e}_j)q_j,\ N_{\text{pool}}\big)$

Initialize $\alpha_i^{(1)} \leftarrow \kappa p_i^0 + 1$, $\beta_i^{(1)} \leftarrow \kappa(1-p_i^0) + 1$, $n_i^+ \leftarrow 0$, $n_i^- \leftarrow 0$ for all $x_i \in \mathcal{D}_{\text{pool}}$

// Online Stage

for $t = 1, \ldots, T$ **do**

  Compute $p_i^{(t)} \leftarrow \alpha_i^{(t)}/(\alpha_i^{(t)} + \beta_i^{(t)})$ and sampling weights w_i via Eq. (6)

  $\mathcal{B}_{\text{eff}} \leftarrow \emptyset$; $\mathcal{P} \leftarrow \mathcal{D}_{\text{pool}}$; $\rho \leftarrow \rho_0$; $c \leftarrow 0$

  while $|\mathcal{B}_{\text{eff}}| < B$ **and** $c < C_{\max}$ **do**

    Sample $\mathcal{B}_{\text{cand}}$ of size $\lceil \rho(B - |\mathcal{B}_{\text{eff}}|) \rceil$ from $\mathcal{P}$ via w (without replacement)

    $\mathcal{P} \leftarrow \mathcal{P} \setminus \mathcal{B}_{\text{cand}}$ ▷ Avoid intra-step re-sampling

    Retain each $x_i \in \mathcal{B}_{\text{cand}}$ w.p. $\gamma_+^{n_i^+}$ if $n_i^+ > 0$; $\gamma_-^{n_i^-}$ if $n_i^- > 0$; else 1

    Add retained prompts to $\mathcal{B}_{\text{eff}}$; compute discard rate $\hat{f}$

    $\rho \leftarrow \max(1.0,\ 1/(1-\hat{f}))$ if $\hat{f} < 1$ else 2ρ; $\quad c \leftarrow c + 1$

  end while

  Truncate $\mathcal{B}_{\text{eff}}$ to size B; perform GRPO update on $\mathcal{B}_{\text{eff}}$

  Update $\alpha_i^{(t+1)}, \beta_i^{(t+1)}, n_i^{\pm}$ from rollout outcomes (Eq. (5))

end for

return π_{θ_T}

Table 7: Performance comparison of various data selection methods. "Ratio" and "RN" denote the ratio of selected data and online rollout numbers, respectively.

Model		Method	Accuracy				Efficiency	
			Math500	AIME25	AMC	Olympiad	Ratio	RN
DeepSeek-R1 -Distill -Qwen-1.5B		Base	54.9	11.9	57.5	40.3	-	-
		Full	79.2	19.6	76.4	50.8	100%	100%
	Offline	PPL-Top	67.8	12.5	59.0	44.5	20%	100%
		PPL-Middle	69.6	13.2	60.8	45.0	20%	100%
		LIMR	70.5	13.8	62.5	46.6	20%	100%
		Learnalign	71.0	14.2	62.9	46.8	20%	100%
		DUDS	**72.8**	**15.6**	**63.6**	**47.5**	20%	100%
	Online	Random	70.2	12.7	64.5	43.8	20%	40%
		Efficient-RL	72.5	13.8	**71.4**	45.2	20%	40%
		GRESO	70.4	15.0	68.9	47.0	20%	40%
		DUDS	**73.7**	**16.8**	70.2	**48.4**	20%	36%

further improves accuracy while reducing the rollout budget to 36%, achieving a more favorable efficiency–performance trade-off than GRESO. These results confirm that the benefits of DUDS generalize across backbones with different architectures and pretraining objectives, demonstrating its robustness as a general-purpose data selection framework for RLVR.

End-to-End Computational Cost. To account for the preprocessing overhead of offline data selection, we report the complete end-to-end training cost on Qwen2.5-Math-1.5B in Table 8. We compare the full DUDS pipeline with GRESO trained on the full dataset and on candidate pools selected by Learnalign or the offline stage of DUDS. Although DUDS incurs a one-time offline cost of 2.4 hours, its reduced online training cost yields the lowest total time of 30.7 hours, compared with 38.5 hours for GRESO on the full dataset, corresponding to a 20.3% reduction. Meanwhile, average

Table 8: End-to-end computational cost comparison on Qwen2.5-Math-1.5B. Total time includes both offline data selection and online training. Online RN denotes the online rollout count relative to the Full baseline, excluding offline rollouts. Accuracy is computed across the four mathematical reasoning benchmarks.

Configuration	Offline (h)	Online (h)	Total (h)	Online RN	Accuracy
GRESO (full dataset)	**0**	38.5	38.5	60%	45.0
Learnalign + GRESO	6.5	34.7	41.2	60%	44.7
DUDS offline + GRESO	2.4	34.7	37.1	60%	47.1
DUDS offline + DUDS online	2.4	**28.3**	**30.7**	**40%**	**47.5**

accuracy improves from 45.0 to 47.5. When both online methods use the same DUDS-selected candidate pool, DUDS further reduces the total time from 37.1 to 30.7 hours while maintaining slightly higher accuracy. These results show that DUDS's efficiency advantage persists even after accounting for offline preprocessing.

Case Study of Asymmetric Zero-Variance Filtering. To better understand the behavioral patterns of DUDS, we present a case study of prompts that were frequently skipped or selected during training on the DAPO-MATH dataset. We categorize the examples into three groups: Frequently Skipped Prompts (Easy), Frequently Selected Prompts, and Frequently Skipped Prompts (Hard), where the moderately difficult prompts that contribute the richest learning signals lie in the middle. We observe that frequently skipped easy prompts typically involve straightforward calculations or routine applications of formulas, so the model solves them correctly across nearly all sampled responses, yielding zero-variance rewards that DUDS filters out via its asymmetric zero-variance mechanism. Frequently selected prompts tend to exhibit moderate difficulty, sitting near the model's current capability boundary, where the Gaussian-shaped sampling weight assigns the highest probability of selection. These prompts contribute more consistently to model improvement throughout training. As for frequently skipped hard prompts, these problems are too challenging for the model to solve even across multiple rollouts, resulting in zero-variance reward signals. Nevertheless, due to the asymmetric design of our filter ($\gamma_- > \gamma_+$), DUDS retains them with greater patience than mastered easy prompts, preserving the opportunity for the model to eventually solve them as its capability grows.

Frequently Skipped Prompts (Easy)

1. **Question:** Two distinct positive integers a and b have integer arithmetic, geometric, and harmonic means. Find the minimal value of $|a - b|$. **Solution:** 30.
2. **Question:** Let a, b, c be the three roots of $x^3 - (k+1)x^2 + kx + 12 = 0$. If $(a-2)^3 + (b-2)^3 + (c-2)^3 = -18$, find k. **Solution:** 5.
3. **Question:** If a, b are real numbers with $a^3+12a^2+49a+69 = 0$ and $b^3-9b^2+28b-31 = 0$, find $a + b$. **Solution:** -1.
4. **Question:** Nonzero real numbers x, y satisfy $x + \frac{1}{y} = 3$ and $y + \frac{1}{x} = 4$. Find $xy + \frac{1}{xy}$. **Solution:** 10.
5. **Question:** Find the smallest positive integer k such that $k+25973$ is a palindrome. **Solution:** 89.

Frequently Selected Prompts

1. **Question:** Suppose integers $a < b < c$ satisfy $a + b + c = 95$ and $a^2 + b^2 + c^2 = 3083$. Find c. **Solution:** 37.

2. **Question:** In triangle ABC, let D be the foot of the altitude from A. Suppose $AD = 4$, $BD = 3$, $CD = 2$, and AB is extended past B to a point E such that $BE = 5$. Determine CE^2. **Solution:** 80.
3. **Question:** Rectangle $HOMF$ has $HO = 11$ and $OM = 5$. Triangle $\triangle ABC$ has orthocenter H and circumcenter O, M is the midpoint of $\overline{BC}$, and altitude $\overline{AF}$ meets $\overline{BC}$ at F. Find the length of BC. **Solution:** 28.
4. **Question:** Find the sum of all possible values of $a + b$ where a, b are nonnegative integers such that $4^a + 2^b + 5$ is a perfect square. **Solution:** 9.
5. **Question:** Triangle ABC is right-angled at A. The circle with center A and radius AB intersects $\overline{BC}$ and $\overline{AC}$ internally at points D and E, respectively. Given that $BD = 20$ and $DC = 16$, determine AC^2. **Solution:** 936.

Frequently Skipped Prompts (Hard)

1. **Question:** There are 10^{2015} planets in an Intergalactic empire. Every two planets are connected by a two-way space line served by one of 2015 travel companies. The Emperor would like to close k of these companies such that it is still possible to reach any planet from any other planet. Find the maximum value of k. **Solution:** 1007.
2. **Question:** Let P be a regular 2006-gon. A diagonal is called *good* if its endpoints divide the boundary of P into two parts, each composed of an odd number of sides of P. The sides of P are also called *good*. Suppose P has been dissected into triangles by 2003 diagonals, no two of which have a common point in the interior of P. Find the maximum number of isosceles triangles having two good sides. **Solution:** 1003.
3. **Question:** Determine the largest k such that for all competitive graphs with 2019 vertices, if the difference between the in-degree and out-degree of any vertex is less than or equal to k, then this graph is strongly connected. **Solution:** 1.
4. **Question:** Triangle $\triangle ABC$ is isosceles with $AB = AC$. Let the incircle of $\triangle ABC$ intersect $\overline{BC}$ and $\overline{AC}$ at D and E, respectively. Let $F \neq A$ be the point such that $DF = DA$ and $EF = EA$. If $AF = 8$ and the circumradius of $\triangle AED$ is 5, find the area of $\triangle ABC$. **Solution:** 40.
5. **Question:** Let $f(n)$ be a function satisfying for all positive integers n: (1) $f(n)$ is a positive integer; (2) $f(n + 1) > f(n)$; (3) $f(f(n)) = 3n$. Find $f(2001)$. **Solution:** 3816.

E Broader Impact

DUDS advances data efficiency in reinforcement learning with verifiable rewards by integrating offline candidate pool construction with online adaptive curriculum learning, offering potential benefits in areas such as automated reasoning and mathematical problem solving. By focusing computation on the most informative training prompts, DUDS reduces the resources required for RLVR training and supports more sustainable development of large language models. We encourage practitioners to evaluate the selection criteria across diverse benchmarks to ensure balanced coverage, fostering efficient and accessible advancement of LLM reasoning capabilities.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We introduce the scope of this paper in the Abstract and the Introduction section, and summarize the contribution in the Introduction and the Conclusion section.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We state the limitation in Section 6 and Section E.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: We include the full set of assumptions and complete proofs in the appendix.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: All datasets and base models we use are publicly available. Our code is provided in an anonymized repository for review.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: All the datasets, code, and model checkpoints are publicly available.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Yes, both in the main paper and in the appendix.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: All experiments are conducted with 3 independent runs using different random seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We provide the information in Section 5.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research conducted in the paper fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Yes, we discuss them in Section E.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We have properly credited the creators and original owners of all assets used in the paper through citations and other appropriate means, ensuring that the licenses and terms of use are respected.

    Guidelines:

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

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

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