LEARNING-ZONE ENERGY: ONLINE DATA SELECTION FOR EFFICIENT RL POST-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning (RL) post-training has emerged as the dominant paradigm for eliciting mathematical reasoning in Large Language Models (LLMs), yet prevailing techniques such as GRPO and DAPO distribute rollout and gradient budgets nearly uniformly across prompts, squandering compute on samples that are already mastered or remain far beyond the model's current capability. To address this fundamental inefficiency, we propose **Learning-Zone Energy** (LZE), a theoretically grounded, fully online data selection framework that concentrates computation on the model's *active learning frontier*. At its core, we define a closed-form *Learning-Zone Energy Score* that combines initial difficulty, current outcome uncertainty, and pass-rate momentum. An analysis of binary group-relative rewards motivates the uncertainty term, while component ablations assess the contribution of the full score. A forward pruner with replay further reduces wall-clock time cost by skipping rollout generation for persistently solved prompts while periodically checking for forgetting. Evaluated on Qwen-family models (1.5B–8B) across GSM8K, MATH and DAPO-MATH, our method retains only 40% of prompt groups for policy updates per step yet matches or surpasses full-data baselines, with especially pronounced out-of-distribution gains on AIME25 ($+45.9\%$) and AMC23 ($+18.2\%$), alongside an estimated 36% reduction in training FLOPs. Our code is available at `https://anonymous.4open.science/r/LZE`.

1 INTRODUCTION

Reinforcement Learning (RL) with verifiable rewards has emerged as the dominant paradigm for improving mathematical reasoning in Large Language Models (Brown et al., 2020; OpenAI et al., 2023). In modern group-based RL pipelines such as GRPO (AI, 2024; 2025) and DAPO (Yu et al., 2025), the model generates multiple rollouts for each prompt, aggregates rewards within each group, and performs policy updates accordingly. However, these methods treat prompt groups nearly uniformly during training, even though the groups are highly heterogeneous in their usefulness: some prompts are already solved reliably and therefore contribute little additional learning signal, whereas others remain far beyond the model's current capability and produce almost no actionable gradient information. Consequently, a considerable fraction of the rollout and optimization budget is spent on prompt groups that are only weakly informative for policy improvement.

To mitigate this inefficiency, prior work has explored curriculum-style training, offline filtering, and online rejection-based heuristics. Early work on curriculum learning and self-paced learning showed that training efficiency can benefit from organizing examples by difficulty (Bengio et al., 2009; Kumar et al., 2010). In the context of LLM post-training, offline filtering and self-training style methods can remove obviously uninformative examples before optimization begins (Singh et al., 2024; Gulcehre et al., 2023). However, such strategies are inherently static and cannot adapt as the policy evolves and the effective difficulty of a prompt changes over time. Online rejection methods are more adaptive, yet they typically apply binary keep-or-drop decisions once a prompt becomes all-correct or all-incorrect (Dong et al., 2023; Xiong et al., 2025), making them insensitive to finer-grained differences in informativeness. A more recent online selection method (Mao et al., 2026) avoids redundant rollouts by predicting prompt solvability via a hidden Markov model, but requires maintaining a learned generative state model, inevitably incurring additional computational overhead. Therefore, existing approaches are often either static, overly rigid, or expensive. What is still missing is a lightweight online criterion that continuously adapts to the current policy, assigns

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

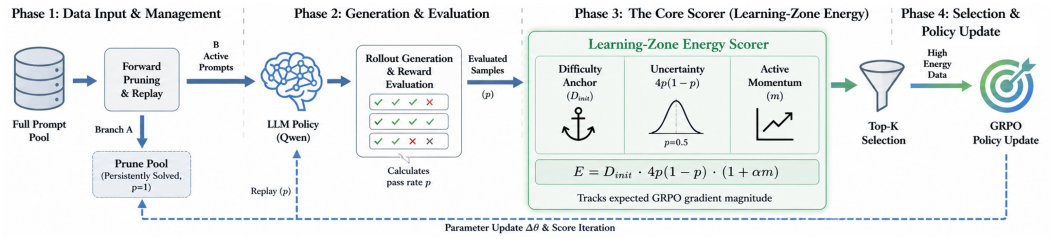


Figure 1: **Overview of the proposed method.** The backward scorer computes the Learning-Zone Energy Score for each prompt group at every training step, guiding Top-K selection for policy updates. The forward pruner tracks group pass rates at the epoch level and skips rollout generation for groups that have been stably solved, with replay providing a safety mechanism to detect forgetting.

prompt groups a graded notion of informativeness, and improves training efficiency without relying on heavy auxiliary machinery.

To address this gap, we propose **Learning-Zone Energy** (LZE), a simple and fully online data selection framework for RL post-training. Our key idea is to prioritise prompt groups in the model's current *learning zone*: groups that are neither already mastered nor completely out of reach, but actively capable of producing meaningful policy updates. Concretely, we define an *Energy Score* that combines three signals: an *initial-difficulty anchor* that preserves attention on historically challenging prompts, a *normalized uncertainty term* based on the current group pass rate that emphasises prompts near the decision boundary, and a *momentum term* that highlights groups on which the policy is actively changing. Our analysis connects the uncertainty term to the empirical variation of centered binary rewards and the probability of obtaining mixed-outcome groups (Section 4.2). A conditional second-moment analysis further characterizes the corresponding policy-gradient signal (Appendix G.1). The momentum term captures changes relative to recent pass-rate history through a causal high-pass filter, and the full score admits an interpretation as a sample-level attention mechanism. To further reduce training cost, we complement the backward scorer with a replay-based *forward pruner* that skips rollout generation for prompts that have remained solved over consecutive epochs, with periodic replay to detect forgetting.

Our contributions are threefold:

- **A principled online data selection framework.** The Learning-Zone Energy Score is a closed-form, per-step criterion that integrates difficulty history, outcome uncertainty, and pass-rate momentum into a single scalar. As it operates solely on group-level pass-rate statistics, LZE is compatible with group-based RL post-training algorithms and requires no modification to the underlying optimizer. Retaining only 40% of prompts per training step, LZE matches and even surpasses full-data baselines across all evaluated configurations, with particularly pronounced out-of-distribution gains: up to $+45.9\%$ on AIME 25 and $+18.2\%$ on AMC 23.
- **Theoretical analysis of gradient informativeness in RL post-training.** We characterize the centered reward variation and the probability of obtaining a non-degenerate group, derive the conditional second-moment structure of a population-centered policy-gradient signal (Theorem 1), and express momentum as a filtered sequence of pass-rate changes.
- **Compute efficiency via dual-stage selection.** The backward selector reduces estimated training FLOPs by 36% under our compute model, as reported in Table 1. The forward pruner provides additional rollout savings, measured separately through wall-clock experiments.

2 Related Work

Curriculum Learning and Data Selection. Curriculum learning (Bengio et al., 2009) and self-paced learning (Kumar et al., 2010) established that organizing examples by difficulty improves optimization. In LLM post-training, data selection strategies prioritizing instances by quality, difficulty, or diversity have been shown to outperform naive quantity scaling (Sorscher et al., 2022; Liu et al., 2024). Difficulty-aware approaches use proxies such as instruction complexity (Lu et al., 2024), model uncertainty (Kuhn et al., 2023), focal weighting (Lin et al., 2017), and online hard-example

mining (Shrivastava et al., 2016) to concentrate the gradient budget on informative samples. Diversity-oriented methods complement these via semantic deduplication, quality-driven subset selection, and gradient-based influence estimation (Abbas et al., 2023; Zhou et al., 2023; Chen et al., 2024; Xia et al., 2024; Xie et al., 2023). Token-level selection further refines this by allocating training signal only to informative tokens within each sequence (Lin et al., 2024). Most of these approaches operate offline and cannot adapt to the evolving policy landscape of online RL.

Data Filtering in RL Post-Training. A growing body of work investigates data filtering during RL post-training, motivated by the observation that not all rollouts contribute equally to policy improvement. Offline pre-filtering removes prompts that are too easy or too hard via a one-time difficulty scoring pass (Singh et al., 2024; Gulcehre et al., 2023), but cannot adapt as the policy evolves. Online rejection methods, including Best-of-N sampling (Cobbe et al., 2021; Yuan et al., 2023; 2024), iterative self-training (Dong et al., 2023; Pang et al., 2024), and Reinforce-Rej (Xiong et al., 2025), operate adaptively but apply binary keep-or-drop decisions that discard the continuous gradient signal. Reward- and signal-driven re-weighting methods use curiosity signals or mistake patterns (Pathak et al., 2017; Wang & Li, 2023); process-level verification provides finer-grained filtering (Uesato et al., 2022; Lightman et al., 2024), while generative reward modeling offers an alternative verifier design (Zhang et al., 2025), all at the cost of auxiliary models. DPS (Mao et al., 2026), a recent method, reduces redundant rollouts by predicting prompt solvability via a hidden Markov model, but necessitates online Bayesian inference and a learned generative state model, inevitably incurring additional overhead. LZE differs from all of the above: the Energy Score is a continuous, closed-form criterion requiring no auxiliary models, and uses current and historical rollout statistics to rank prompt groups.

Additional related work on reinforcement learning for mathematical reasoning is discussed in Appendix A.

3 Background

We review the group-based RL post-training paradigm, introduce the notation used throughout, and identify the gradient-informativeness property that motivates our approach.

RL post-training for mathematical reasoning. Given a prompt dataset $\mathcal{D} = \{x_i\}_{i=1}^{N}$ and a policy π_θ, RL post-training maximises the expected reward:

$$\max_{\theta} \; \mathbb{E}_{x\sim\mathcal{D},\; y\sim\pi_\theta(\cdot|x)}\big[r(x,y)\big], \tag{1}$$

where $r(x, y) \in \{0, 1\}$ is a rule-based binary verifier that checks the correctness of response y for prompt x. This binary reward structure, standard in mathematical reasoning, makes the per-prompt *pass rate* a natural proxy for the policy's current capability on that problem.

Group Relative Policy Optimization (GRPO). GRPO (AI, 2024; 2025) optimizes Eq. (1) without a learned value network, using within-group reward comparisons. Our implementation centers rewards by the group mean without dividing by the group standard deviation. For prompt x_i, the policy generates n i.i.d. rollouts $y_{i,1}, \ldots, y_{i,n} \sim \pi_\theta(\cdot|x_i)$ with binary rewards $r_{i,k} = r(x_i, y_{i,k})$. The empirical pass rate $p_i^{(t)} = n^{-1}\sum_k r_{i,k}$ serves as the group baseline, and the centered group-relative advantage is

$$\tilde{A}_{i,k} \;=\; r_{i,k} - p_i^{(t)}. \tag{2}$$

GRPO then maximizes the clipped surrogate objective

$$\mathcal{L}_{\mathrm{GRPO}}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{n}\sum_{k=1}^{n}\min\left(\frac{\pi_\theta(y_{i,k}\mid x_i)}{\pi_{\theta_{\mathrm{old}}}(y_{i,k}\mid x_i)}\tilde{A}_{i,k},\; \mathrm{clip}\left(\frac{\pi_\theta(y_{i,k}\mid x_i)}{\pi_{\theta_{\mathrm{old}}}(y_{i,k}\mid x_i)}, 1{-}\epsilon, 1{+}\epsilon\right)\tilde{A}_{i,k}\right), \tag{3}$$

where the clipped importance ratio prevents π_θ from deviating excessively from the behaviour policy $\pi_{\theta_{\mathrm{old}}}$ (Schulman et al., 2017). DAPO (Yu et al., 2025) removes the KL-divergence penalty from the reference policy and adopts a token-level entropy-aware clipping to further stabilize training.

Gradient informativeness and the learning zone. At the rollout policy, where the importance ratio is one and clipping is inactive, the policy-gradient contribution from prompt i takes the form below, with the sampled advantages held fixed during differentiation:

$$\nabla_\theta \mathcal{L}^{(i)} \;\propto\; \frac{1}{n}\sum_{k=1}^{n} \tilde{A}_{i,k}\, \nabla_\theta \log \pi_\theta(y_{i,k} \mid x_i). \tag{4}$$

When $p_i^{(t)} = 1$ or $p_i^{(t)} = 0$, every centered advantage is zero and the group contributes no policy-gradient update. Mixed-outcome groups provide nonzero centered rewards, whose empirical second moment is $p_i^{(t)}(1 - p_i^{(t)})$. This quantity peaks at 0.5, motivating the *learning zone* as a regime with substantial within-group reward variation. Section 4.2 formalizes this connection.

Online data selection. At each training step t, the *online data selection problem* is to identify a subset $\mathcal{S}_t \subseteq \mathcal{D}$ with $|\mathcal{S}_t| = \lfloor \kappa N \rfloor$ ($\kappa \in (0, 1)$) that maximizes expected policy improvement per unit of compute, using only information available at step t. A practical criterion must be (i) updated online without offline pre-processing, (ii) aligned with the gradient signal characterised by Eq. (4), and (iii) computationally cheap, requiring no auxiliary models beyond the standard rollout phase.

4 METHODOLOGY

This section presents the LZE framework in three parts. We introduce the Learning-Zone Energy Score and its three components (Section 4.1), analyze centered reward variation and the temporal structure of momentum (Section 4.2), and describe backward Top-K selection and forward pruning with replay (Section 4.3).

4.1 THE LEARNING-ZONE ENERGY SCORE

The key challenge in online data selection for RL post-training is to construct a criterion that adapts to the policy's evolving capability. Motivated by the centered reward variation in Section 3, we combine three aspects of prompt informativeness: initial difficulty, current outcome uncertainty, and recent learning dynamics. For prompt i at training step t, we define the **Learning-Zone Energy Score** as

$$E_i^{(t)} \;=\; \underbrace{D_i^{(0)}}_{\text{difficulty anchor}} \cdot \underbrace{4p_i^{(t)}\left(1 - p_i^{(t)}\right)}_{\text{outcome uncertainty}} \cdot \underbrace{\left(1 + \alpha\, m_i^{(t)}\right)}_{\text{momentum}}, \tag{5}$$

where the three multiplicative factors correspond to the three signals above and each plays a distinct and non-redundant role.

Outcome uncertainty: $4p_i^{(t)}(1 - p_i^{(t)})$. The central term is the normalized Bernoulli variance of the current pass rate, which lies in $[0, 1]$, peaks at $p = 0.5$, and vanishes at both extremes. This term directly implements the learning-zone intuition: it concentrates selection on prompt groups where outcomes are mixed, i.e., neither consistently correct nor consistently incorrect, and suppresses both extremes. Section 4.2 relates this term to centered reward variation and the probability of obtaining a mixed-outcome group. $4p(1 - p)$ is *symmetric* in the pass rate and down-weights both extremes.

Difficulty anchor: $D_i^{(0)}$. Prompts with the same current pass rate can have different learning histories. We record the pass rate at the start of training and set $D_i^{(0)} = 1 - p_i^{(0)} \in [0, 1]$, giving a fixed prior over difficulty under the initial policy. This anchor favors initially challenging prompts over initially easy prompts whose current pass rates have temporarily decreased.

Pass-rate momentum: $m_i^{(t)}$. We track recent learning dynamics via an exponential moving average $\mu_i^{(t)} = \lambda \mu_i^{(t-1)} + (1 - \lambda) p_i^{(t)}$ with decay $\lambda \in (0, 1)$, and define the momentum $m_i^{(t)} = p_i^{(t)} - \mu_i^{(t-1)}$. Positive momentum raises the score, negative momentum lowers it, and zero momentum leaves it unchanged. With the default $\alpha = 0.3$, the multiplier $(1 + \alpha m_i)$ lies in $[0.7, 1.3]$. Section 4.2 characterizes this signed change signal as a causal high-pass filter on the pass-rate sequence.

4.2 Analysis of the Uncertainty and Momentum Terms

Group-relative reward signal. For the centered advantages in Eq. (2), binary rewards give the exact identity

$$\frac{1}{n}\sum_{k=1}^{n}\tilde{A}_{i,k}^{2} = p_i^{(t)}\big(1-p_i^{(t)}\big). \tag{6}$$

Thus, the uncertainty term in Eq. (5) is four times the mean squared centered advantage: it is zero for all-correct or all-incorrect groups and largest for equally mixed outcomes. Let $p_i = \Pr_{y\sim\pi_\theta(\cdot|x_i)}(r(x_i,y)=1)$ denote the true pass rate under the current policy, distinct from the empirical rate $p_i^{(t)}$. For $n \geq 2$ independent rollouts,

$$\Pr\Big(0 < p_i^{(t)} < 1\Big) = 1 - p_i^n - (1-p_i)^n. \tag{7}$$

This probability also peaks at $p_i = 0.5$, motivating selection in the intermediate pass-rate regime where groups are most likely to provide nonzero relative reward signals. Appendix G gives the corresponding score-function second-moment decomposition and its relation to $p_i(1-p_i)$.

Momentum as a temporal change signal. With $\mu_i^{(0)} = p_i^{(0)}$ and $\Delta p_i^{(t)} = p_i^{(t)} - p_i^{(t-1)}$, the momentum satisfies

$$m_i^{(t)} = \Delta p_i^{(t)} + \lambda m_i^{(t-1)} = \sum_{s=1}^{t}\lambda^{t-s}\Delta p_i^{(s)}, \qquad m_i^{(0)} = 0. \tag{8}$$

It therefore accumulates recent pass-rate changes with exponentially decaying weights. Positive momentum raises a prompt's score, negative momentum lowers it, and zero momentum leaves the uncertainty and difficulty factors unchanged. The difficulty anchor and momentum combine this current reward signal with initial difficulty and recent learning history; their contributions are evaluated in the component ablation.

Interpretation via attention. The Energy Score can be read, by analogy, as an unnormalized sample-level attention weight: outcome uncertainty identifies groups with relative reward variation, while the difficulty anchor and momentum modulate their priority using learning history. Top-K selection then allocates policy updates to the retained groups. Compared with Focal Loss (Lin et al., 2017) and OHEM (Shrivastava et al., 2016), this rule combines a symmetric uncertainty term with temporally varying prompt statistics.

4.3 Prompt Selection and Training Algorithm

Backward selection. At each training step t, LZE first collects rollouts for all prompts in the active pool, computes pass rates and Energy Scores via Eq. (5), and then retains the top-κ fraction for policy optimization. To balance exploitation with exploration, small Gumbel perturbations are added before ranking: $\tilde{E}_i = E_i^{(t)} + \tau\xi_i$, $\xi_i \overset{\text{i.i.d.}}{\sim} \text{Gumbel}(0,1)$, with $\tau = 0.05$. The top-$K = \lfloor\kappa|\mathcal{A}|\rfloor$ prompts by perturbed score form $\mathcal{S}_t$; only these groups participate in policy gradient updates. Appendix F.1 analyzes this perturbation and reports its ablation. Backward selection mainly reduces *optimization cost*, and rollouts are still collected over the full active pool to update Energy Scores.

Forward pruning with replay. At the epoch level, if prompt i achieves an observed pass rate of 1 for $T_{\text{prune}} = 2$ consecutive epochs, it is moved to a prune pool $\mathcal{P}$ and skipped in future rollout generation. At the start of each epoch, a fraction ρ of $\mathcal{P}$ is re-evaluated, and any prompt that no longer achieves full correctness is restored to the active pool. The two stages are complementary: the backward selector concentrates the *gradient* budget on the most informative groups, while the forward pruner eliminates *rollout* cost for persistently solved prompts. The complete training procedure is presented in Algorithm 1.

5 Experiments

We evaluate LZE on mathematical reasoning benchmarks across multiple model scales, comparing against representative data selection methods spanning offline filtering, online and SFT-based rejection.

Algorithm 1 Learning-Zone Energy Training

Require: Dataset $\mathcal{D}$, policy π_{θ_0}, selection ratio κ, EMA decay λ, momentum weight α, prune threshold T_{prune}, replay ratio ρ

Initialise: Run a single forward pass over $\mathcal{D}$; for each i, compute $p_i^{(0)}$, set $D_i^{(0)} \leftarrow 1 - p_i^{(0)}$, $\mu_i^{(0)} \leftarrow p_i^{(0)}$, epoch-correct counter $c_i \leftarrow 0$
Active pool $\mathcal{A} \leftarrow \mathcal{D}$; Prune pool $\mathcal{P} \leftarrow \emptyset$
for each epoch $e = 1, 2, \ldots$ **do**
  Replay: Sample $\lfloor \rho |\mathcal{P}| \rfloor$ prompts from $\mathcal{P}$; run rollouts; move any with an observed pass rate below 1 back to $\mathcal{A}$
  for each training step t within epoch e **do**
    Collect rollouts for all $i \in \mathcal{A}$; compute $p_i^{(t)}$
    $\mu_i^{(t)} \leftarrow \lambda\, \mu_i^{(t-1)} + (1-\lambda)\, p_i^{(t)}$ ▷ EMA update
    $m_i^{(t)} \leftarrow p_i^{(t)} - \mu_i^{(t-1)}$ ▷ Momentum
    $E_i^{(t)} \leftarrow D_i^{(0)} \cdot 4p_i^{(t)}(1-p_i^{(t)}) \cdot (1 + \alpha\, m_i^{(t)})$ ▷ Energy Score
    $\mathcal{S}_t \leftarrow \text{Gumbel-Top-K}\big(\{E_i^{(t)}\}_{i\in\mathcal{A}}, \lfloor \kappa |\mathcal{A}| \rfloor\big)$ ▷ Backward selection
    $\theta_{t+1} \leftarrow \text{GradStep}(\pi_{\theta_t}, \mathcal{S}_t)$ ▷ Policy update on selected subset
  end for
  Forward pruning: For each $i \in \mathcal{A}$, increment c_i if $p_i^{(t_{\text{last}})} = 1$, else reset $c_i \leftarrow 0$; if $c_i \geq T_{\text{prune}}$, move i from $\mathcal{A}$ to $\mathcal{P}$
end for

We further analyze training efficiency and convergence dynamics and conduct ablation studies to verify the contribution of each Energy Score component and the sensitivity to key hyperparameters.

Models and Datasets. We evaluate our method on mathematical reasoning, where models must produce step-by-step Chain-of-Thought (CoT) reasoning together with a boxed final answer. To ensure broad coverage, we experiment with models from the Qwen2.5 (Yang et al., 2024a) and Qwen3 (Yang et al., 2025) families (the latter with thinking mode enabled throughout training), and the full list of evaluated variants is given in Table 1. Ablation studies are primarily conducted with Qwen2.5-Math-1.5B and Qwen2.5-Math-7B (Yang et al., 2024b), chosen for their strong performance ceiling and stable inference behaviour, making them reliable starting points for post-training. We train on three datasets of increasing difficulty: GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), and DAPO-MATH (English split) (Yu et al., 2025).

Hyper-parameters. We follow the default configurations of the Verl framework (Sheng et al., 2025) for PPO and GRPO training. Following recent practice in group-based RL post-training (Yu et al., 2025; He et al., 2025), we remove the KL-divergence penalty and use temperature $= 1.0$, top-$p = 1.0$, and top-$k = -1$ to encourage diverse rollout generation. All runs use AdamW with learning rate 1×10^{-6}. The prompt batch size is 1024 for models under 4B parameters and 512 for larger models, with $n = 8$ rollouts per prompt. Maximum generation length is 4096 tokens for Qwen2.5-family models and 16384 tokens for Qwen3-family models. The selection ratio κ=0.4 and momentum coefficient $\alpha = 0.3$ are fixed across all main experiments, and sensitivity analyses are in Section 5.2.

Evaluation. We use a standardized chat template (Appendix B) and determine correctness via rule-based answer matching using `math-verify`. We report Pass@1 accuracy on both in-domain and out-of-distribution (OOD) benchmarks, together with estimated FLOPs budgets, wall-clock training time, and convergence speed.

5.1 Main Results

Accuracy and out-of-distribution generalization. Table 1 summarizes the Pass@1 accuracy across both in-domain and out-of-distribution (OOD) mathematical benchmarks. We report *Base* as the strong baseline without data filtering. Our method consistently outperforms the full-data baseline in nearly all configurations while training on only 40% of prompts per step, confirming that

Table 1: Main Performance & Efficiency. Pass@1 comparison on in-domain and OOD evaluation sets, alongside estimated computational savings.

Training Set	Model		Accuracy (Pass@1)						FLOPs Budget
			GSM8K	MATH	DAPO-MATH	AMC23	AIME25	Avg	
GSM8K	Qwen2.5-Math-1.5B	Base	0.859	0.538	0.249	0.575	0.111	0.466	5.47×10^{19}
		Ours	**0.860**	**0.553**$^{\uparrow +2.8\%}$	**0.265**$^{\uparrow +6.4\%}$	**0.600**$^{\uparrow +4.3\%}$	**0.122**$^{\uparrow +9.9\%}$	**0.480**$^{\uparrow +2.9\%}$	3.50×10^{19}
	Qwen3-1.7B	Base	0.904	0.573	0.450	0.600	0.211	0.548	4.94×10^{19}
		Ours	**0.912**	**0.578**	**0.452**	**0.700**$^{\uparrow +16.7\%}$	**0.267**$^{\uparrow +26.5\%}$	**0.582**$^{\uparrow +6.2\%}$	3.17×10^{19}
MATH	Qwen2.5-Math-1.5B	Base	0.807	**0.583**	0.281	0.550	0.122	0.469	5.82×10^{19}
		Ours	**0.825**$^{\uparrow +2.2\%}$	**0.583**	**0.310**$^{\uparrow +10.3\%}$	**0.650**$^{\uparrow +18.2\%}$	**0.178**$^{\uparrow +45.9\%}$	**0.509**$^{\uparrow +8.6\%}$	3.73×10^{19}
	Qwen2.5-Math-7B	Base	0.868	0.621	0.440	0.700	0.278	0.581	1.70×10^{20}
		Ours	0.876	**0.628**	**0.462**$^{\uparrow +5.0\%}$	**0.725**$^{\uparrow +3.6\%}$	**0.311**$^{\uparrow +11.9\%}$	**0.600**$^{\uparrow +3.2\%}$	1.09×10^{20}
	Qwen3-4B	Base	**0.945**	0.771	0.576$^{\downarrow}$	0.875	0.544	0.742	1.18×10^{20}
		Ours	0.942	**0.775**	0.576$^{\downarrow}$	**0.900**$^{\uparrow +2.9\%}$	**0.556**$^{\uparrow +2.2\%}$	**0.750**$^{\uparrow +1.0\%}$	7.55×10^{19}
DAPO-MATH	Qwen2.5-Math-7B	Base	**0.928**	0.699	0.530	0.775	0.344	0.655	2.94×10^{20}
		Ours	**0.928**	**0.708**$^{\uparrow +1.3\%}$	**0.546**$^{\uparrow +3.0\%}$	**0.825**$^{\uparrow +6.5\%}$	**0.356**$^{\uparrow +3.5\%}$	**0.673**$^{\uparrow +2.7\%}$	1.89×10^{20}
	Qwen3-4B	Base	0.914	0.630	**0.807**	**0.925**	0.544	0.764	8.56×10^{19}
		Ours	**0.941**$^{\uparrow +3.0\%}$	**0.639**$^{\uparrow +1.4\%}$	0.795	0.900	**0.556**$^{\uparrow +2.2\%}$	**0.766**$^{\uparrow +0.3\%}$	5.48×10^{19}
	Qwen3-8B	Base	0.906	0.626	0.768	0.825	0.456	0.716	9.85×10^{19}
		Ours	**0.911**	**0.630**	**0.807**$^{\uparrow +5.1\%}$	**0.925**$^{\uparrow +12.1\%}$	**0.500**$^{\uparrow +9.6\%}$	**0.755**$^{\uparrow +5.4\%}$	6.30×10^{19}

concentrating the gradient budget on the learning frontier is more effective than uniform allocation. The gains are most pronounced on the challenging OOD benchmarks AMC 23 and AIME 25, which measure transferable mathematical reasoning rather than in-distribution recall. Representative gains include: Qwen2.5-Math-1.5B on MATH achieving $+45.9\%$ on AIME 25 and $+18.2\%$ on AMC 23; Qwen3-1.7B on GSM8K achieving $+26.5\%$ on AIME 25 and $+16.7\%$ on AMC 23; and Qwen3-8B on DAPO-MATH achieving $+12.1\%$ on AMC 23 and $+9.6\%$ on AIME 25. These OOD improvements are consistent with the theoretical prediction of Theorem 1: by suppressing both trivially solved and completely failed prompt groups, LZE prevents the policy from overfitting to the in-distribution difficulty profile, driving capability gains that transfer more broadly. The gains are particularly pronounced for smaller models (1.5B and 1.7B), where the learning zone is narrower and the Energy Score's discriminative power is highest. Furthermore, consistent improvements across all scales from 1.5B to 8B and across both the Qwen2.5 and Qwen3 families confirm that learning-zone targeting is broadly applicable and not tied to any particular architecture or capacity regime.

Efficiency and training progress. The proposed data selection method reduces theoretical training FLOPs by approximately 36% by restricting backpropagation to the top-40% fraction of prompt groups, while maintaining full rollout coverage to recompute Energy Scores at each step (see Appendix E for the derivation). Figure 2 plots Pass@1 against wall-clock training time. For Qwen2.5-Math-1.5B, the baseline stabilizes at 3.97 hours; LZE reaches better performance in 2.40 hours ($1.65\times$ faster). For the 7B model, convergence is reached in 3.51 hours versus 4.92 hours for the baseline ($1.40\times$ faster). Adding the forward pruner yields further speedups of $1.80\times$ and $1.91\times$ for the 1.5B and 7B models, respectively, at a modest accuracy trade-off relative to the backward-only variant. Notably, uniform sampling at the same 40% retention ratio consistently underperforms the full-data baseline and even falls below the forward pruner in final accuracy, confirming that the gains of LZE stem from the principled identification of learning-zone prompts rather than from any implicit regularization effect of reduced data volume.

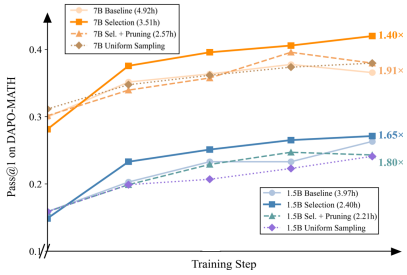


Figure 2: Pass@1 vs. wall-clock time for Qwen2.5-Math models. The $\times$ annotations denote speedup multipliers relative to the full-data baseline.

Comparison with existing methods. Table 2 compares LZE against offline, online, and SFT-based baselines on the MATH dataset, and empirical results suggest:

Table 2: Comparison of Pass@1 performance between the proposed method and existing data filtering baselines trained on MATH. Avg Score is the mean over DAPO-MATH, AMC23, and AIME25. "rem." stands for remove.

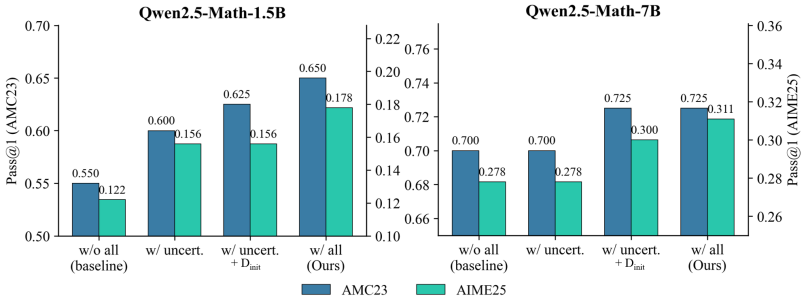
Method	Qwen2.5-Math-1.5B				Qwen2.5-Math-7B			
	DAPO-MATH	AMC23	AIME25	Avg Score	DAPO-MATH	AMC23	AIME25	Avg Score
Baseline	0.281	0.550	0.122	0.318	0.440	0.700	0.278	0.473
Ours	0.310	**0.650**	**0.178**	**0.379**	**0.462**	**0.725**	**0.311**	**0.500**
Reinforce-Rej (with online filtering)	0.235	0.525	0.122	0.294	0.351	**0.725**	0.200	0.425
Pre-Filter (rem. all wrong)	**0.327**	0.575	0.122	0.341	0.447	0.675	0.244	0.455
Pre-Filter (rem. all right)	0.305	0.575	0.167	0.349	0.458	0.675	0.279	0.471
Pre-Filter (discard all)	0.303	0.600	0.156	0.353	0.428	**0.725**	0.244	0.466
RAFT	0.281	0.575	0.133	0.330	0.363	0.575	0.189	0.376
Uniform sampling	0.273	0.550	0.122	0.315	0.437	0.675	0.289	0.467

- *Online selection adapts to the evolving policy.* The Pre-Filter variants improve the average score on the 1.5B model but remain below the baseline on the 7B model. These filters select data using a snapshot of model capability at initialization. LZE recomputes its Energy Score at every training step and obtains a higher average score than these static filters on both backbones.

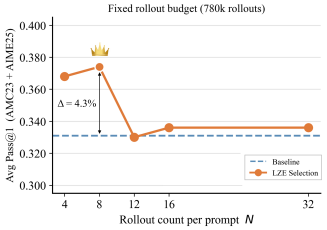
- *Graded ranking improves on binary rejection.* Reinforce-Rej (Xiong et al., 2025) uses a binary filtering rule. LZE instead ranks prompt groups with a continuous Energy Score before Top-K selection and achieves a higher average score on both backbones.

- *Selection for policy updates complements RL training.* RAFT (Dong et al., 2023) retains positively rewarded rollouts for supervised training. LZE retains group-relative policy updates and achieves a higher average score than RAFT on both backbones.

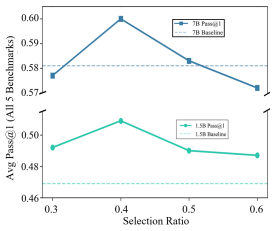
Additional comparisons with recent online selection methods are reported in Appendix F.3.



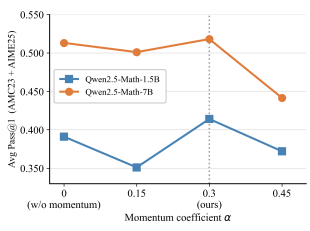
(a) Component ablation.



(b) Rollout sensitivity.



(c) Selection-ratio sensitivity.



(d) Momentum sensitivity.

Figure 3: Ablation studies on Energy Score components, rollout count n, selection ratio κ, and momentum coefficient α, conducted on Qwen2.5-Math models trained on MATH.

5.2 Ablation Study

Contribution of each Energy Score component. We ablate each factor in Eq. (5) by removing it in turn and measuring Pass@1 on AMC23 and AIME25 using Qwen2.5-Math-1.5B and Qwen2.5-Math-7B. As shown in Figure 3a, removing the uncertainty term $4p(1-p)$ causes the largest degradation, supporting its central role in the selection rule. Removing the difficulty anchor $D_i^{(0)}$ or the momentum term also reduces OOD performance, demonstrating the incremental value of initial difficulty and recent learning dynamics.

Sensitivity to rollout n. Under a fixed total rollout budget of 780k (corresponding to n=8 over approximately 7 epochs), we vary the per-prompt rollout $n \in \{4, 8, 12, 16, 32\}$ for both the baseline and LZE, holding all other hyperparameters constant. As shown in Figure 3b, LZE consistently outperforms the baseline across all values of n, with peak absolute performance at n=8. Smaller n reduces the number of samples used to estimate the group-level pass rate, making the Energy Score noisy and less discriminative. Larger n improves per-prompt accuracy but reduces the number of epochs within the fixed budget, limiting how many times the score can adapt to policy changes. The default n=8 strikes the best balance between estimate quality and adaptation frequency, and the performance gap between LZE and the baseline is most pronounced at this setting.

Sensitivity to selection ratio κ. Figure 3c reports the averaged performance under varying κ across both the 1.5B and 7B models, and Pass@1 peaks at $\kappa = 0.4$. Lower ratios leave fewer groups for each policy update, while higher ratios admit more low-energy groups. All-correct and all-incorrect groups contribute zero centered reward signal by Eq. (6); the selection ratio balances update coverage with concentration on mixed-outcome groups.

Sensitivity to the momentum coefficient α. The momentum term $(1 + \alpha m_i^{(t)})$ modulates the Energy Score using recent pass-rate changes. Setting $\alpha = 0$ removes momentum while retaining the difficulty anchor and uncertainty term. We sweep $\alpha \in \{0, 0.15, 0.3, 0.45\}$ on MATH with Qwen2.5-Math-1.5B and Qwen2.5-Math-7B, reporting Avg Pass@1 = (AMC23 + AIME25) / 2.

Figure 3d reveals a consistent non-monotone pattern across both model scales. Setting $\alpha = 0$ (no momentum) already yields strong performance, while the momentum signal provides a meaningful further boost at $\alpha = 0.3$: +0.023 for 1.5B (0.391→0.414) and +0.005 for 7B (0.513→0.518). A smaller $\alpha = 0.15$ under-utilises the temporal change signal and falls below the no-momentum baseline for the 1.5B model, suggesting that weak momentum can transiently mislead selection before the EMA has time to smooth out. An excessively large $\alpha = 0.45$ makes the score overly sensitive to single-step fluctuations in the pass rate: transient improvements are over-rewarded, causing the selector to repeatedly re-select prompts that exhibited one lucky run but have not yet stably improved, which degrades the training distribution and reduces final accuracy. The sweet spot at $\alpha = 0.3$ strikes the best balance between responsiveness and stability across both model sizes, and we adopt it as the default for all experiments in the main paper.

6 Conclusion

In this work, we propose **Learning-Zone Energy (LZE)**, a fully online data selection framework for efficient RL post-training. The central insight is that intermediate pass rates provide substantial centered reward variation and a high probability of mixed-outcome groups. Building on this analysis, the Learning-Zone Energy Score fuses *initial difficulty*, *outcome uncertainty*, and *pass-rate momentum* into a closed-form criterion updated at every training step without auxiliary models. Empirical results demonstrate that LZE improves performance over the full-data baseline on nearly all benchmarks while allocating policy updates to only 40% of prompts per step, with particularly pronounced gains on OOD settings. One limitation of the current study is that all experiments focus on mathematical reasoning tasks with binary verifiable rewards, so whether the same scoring rule transfers to open-ended or noisy-reward settings remains an important direction for future work.

7 AI USE STATEMENT

In this work, we used generative AI tools for assisting in the writing of proofs, cleaning and reformatting dataset and grammatical refinement toward this manuscript. We have not used generative AI tools for generating synthetic data sets, proposing or refining hypotheses, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, supporting qualitative and thematic data analysis and interpreting results. Additionally, we used generative AI tools for editing software code (debugging), summarizing or analyzing existing literature, discovering research topics or identifying gaps, brainstorming, sourcing/searching for information, editing a research paper to improve readability, identifying relevant literature, formatting references. We have reviewed all AI-assisted work. To be specific, LLM-generated code was verified and tested for correctness and during grammatically refining this manuscript using AI tools, we manually checked contents generated by AI and substantially changed AI-written text. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

8 ETHICS STATEMENT

The potential negative social impacts of our method are minor, and align with those typically associated with general LLM reasoning technologies. We emphasise the importance of adhering to the principles of fair and safe deployment of LLM systems, including careful consideration of the training data, transparency about model capabilities and limitations, and ongoing monitoring for unintended consequences.

9 REPRODUCIBILITY STATEMENT

The paper fully discloses all information required to reproduce its main experimental results that underpin the core claims and conclusions of the work. For the proposed method and all baselines, the paper provides complete and clear descriptions of the experimental design, evaluation protocols, and all necessary implementation details to enable independent replication of the results, in full compliance with ICLR reproducibility requirements.

REFERENCES

Amro Abbas, Kushal Tirumala, Daniel Simig, Surya Ganguli, and Ari S. Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023. URL https://arxiv.org/abs/2303.09540.

Deepseek AI. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Deepseek AI. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Ahmadian Arash, Cremer Chris, Gallé Matthias, Fadaee Marzieh, Kreutzer Julia, Pietquin Olivier, Üstün Ahmet, and Hooker Sara. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267. Association for Computational Linguistics, August 2024. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, 2009. ISBN 9781605585161. doi: 10.1145/1553374.1553380. URL https://doi.org/10.1145/1553374.1553380.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data. In *International Conference on Learning Representations (ICLR)*, pp. 34767–34797, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/9543942c237ded1b39b1fd37259ff88e-Paper-Conference.pdf.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, KaShun SHUM, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://arxiv.org/abs/2304.06767.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. Reinforced self-training (rest) for language modeling, 2023. URL https://arxiv.org/abs/2308.08998.

Bingxiang He, Zekai Qu, Zeyuan Liu, Yinghao Chen, Yuxin Zuo, Cheng Qian, Kaiyan Zhang, Weize Chen, Chaojun Xiao, Ganqu Cui, Ning Ding, and Zhiyuan Liu. JustRL: Scaling a 1.5b LLM with a simple RL recipe, 2025. URL https://arxiv.org/abs/2512.16649.

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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper-round2.pdf.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! In *Deep Reinforcement Learning Meets Structured Prediction, ICLR 2019 Workshop, New Orleans, Louisiana, United States, May 6, 2019*. OpenReview.net, 2019a. URL https://openreview.net/forum?id=r1lgTGL5DE.

Wouter Kool, Herke van Hoof, and Max Welling. Stochastic beams and where to find them: The Gumbel-Top-k trick for sampling sequences without replacement. In *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2019b. URL https://proceedings.mlr.press/v97/kool19a.html.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

M. Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 23, pp. 1189–1197, 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/e57c6b956a6521b28495f2886ca0977a-Paper.pdf.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, pp. 39578–39601, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/aca97732e30bcf1303bc22ac3924fd16-Paper-Conference.pdf.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2999–3007, 2017. doi: 10.1109/ICCV.2017.324.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Not all tokens are what you need for pretraining. In *Advances in Neural Information Processing Systems*, volume 37, pp. 29029–29063, 2024. doi: 10.52202/079017-0914. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/3322a9a72a1707de14badd5e552ff466-Paper-Conference.pdf.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *International Conference on Learning Representations (ICLR)*, pp. 22353–22373, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/6091f2bb355e960600f62566ac0e2862-Paper-Conference.pdf.

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, 2025. URL https://arxiv.org/abs/2503.20783.

Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. #instag: Instruction tagging for analyzing supervised fine-tuning of large language models. In *International Conference on Learning Representations (ICLR)*, pp. 36456–36474, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/9dae2a90bae49dc874ce1ca8fcc20879-Paper-Conference.pdf.

Yixiu Mao, Yun Qu, Cheems Wang, Heming Zou, and Xiangyang Ji. Dynamics-Predictive Sampling for Active RL Finetuning of Large Reasoning Models. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=voeheZjd8p.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini,

Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2023. URL https://arxiv.org/abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative Reasoning Preference Optimization. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 116617–116637. Curran Associates, Inc., 2024. doi: 10.52202/079017-3702. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/d37c9ad425fe5b65304d500c6edcba00-Paper-Conference.pdf.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 488–489, 2017. doi: 10.1109/CVPRW.2017.70.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297. ACM, March 2025. doi: 10.1145/3689031.3696075. URL http://dx.doi.org/10.1145/3689031.3696075.

Taiwei Shi, Yiyang Wu, Linxin Song, Tianyi Zhou, and Jieyu Zhao. Efficient reinforcement finetuning via adaptive curriculum learning, 2025. URL https://arxiv.org/abs/2504.05520.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 761–769, June 2016.

Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, Aaron T Parisi, Abhishek Kumar, Alexander A

Alemi, Alex Rizkowsky, Azade Nova, Ben Adlam, Bernd Bohnet, Gamaleldin Fathy Elsayed, Hanie Sedghi, Igor Mordatch, Isabelle Simpson, Izzeddin Gur, Jasper Snoek, Jeffrey Pennington, Jiri Hron, Kathleen Kenealy, Kevin Swersky, Kshiteej Mahajan, Laura A Culp, Lechao Xiao, Maxwell Bileschi, Noah Constant, Roman Novak, Rosanne Liu, Tris Warkentin, Yamini Bansal, Ethan Dyer, Behnam Neyshabur, Jascha Sohl-Dickstein, and Noah Fiedel. Beyond human data: Scaling self-training for problem-solving with language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://arxiv.org/abs/2312.06585.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 19523–19536, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/7b75da9b61eda40fa35453ee5d077df6-Paper-Conference.pdf.

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. URL https://neurips.cc/virtual/2025/poster/115452.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback, 2022. URL https://arxiv.org/abs/2211.14275.

Danqing Wang and Lei Li. Learning from mistakes via cooperative study assistant for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 10667–10685, December 2023. doi: 10.18653/v1/2023.emnlp-main.659. URL https://aclanthology.org/2023.emnlp-main.659/.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3-4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Forty-first International Conference on Machine Learning*, Proceedings of Machine Learning Research, pp. 54104–54132, 2024. URL https://proceedings.mlr.press/v235/xia24c.html.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34201–34227, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6b9aa8f418bde2840d5f4ab7a02f663b-Paper-Conference.pdf.

Wei Xiong, Jiarui Yao, Yuhui Xu, Bo Pang, Lei Wang, Doyen Sahoo, Junnan Li, Nan Jiang, Tong Zhang, Caiming Xiong, and Hanze Dong. A minimalist approach to LLM reasoning: from rejection sampling to reinforce, 2025. URL https://arxiv.org/abs/2504.11343.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang,

Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2024a. URL https://arxiv.org/abs/2412.15115.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement, 2024b. URL https://arxiv.org/abs/2409.12122.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/271db9922b8d1f4dd7aaef84ed5ac703-Paper-Conference.pdf.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, 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 *Advances in Neural Information Processing Systems*, volume 38, pp. 113222–113244, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a4277440d50f1f15d2cb4c14f7e0c0d2-Paper-Conference.pdf.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2024. URL https://arxiv.org/abs/2401.10020.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models, 2023. URL https://arxiv.org/abs/2308.01825.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 15476–15488. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/639a9a172c044fbb64175b5fad42e9a5-Paper-Conference.pdf.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Seyed Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *International Conference on Learning Representations (ICLR)*, pp. 12476–12505, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/214308a2d5e3f83ef9ad2739e1cbc46d-Paper-Conference.pdf.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. Lima: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, pp. 55006–55021, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/ac662d74829e4407ce1d126477f4a03a-Paper-Conference.pdf.

A Additional Related Work

Reinforcement Learning for Mathematical Reasoning. Building on chain-of-thought reasoning (Wei et al., 2022), self-consistency decoding (Wang et al., 2023), and tree search (Yao et al., 2023; Zelikman et al., 2022), reinforcement learning has become the dominant post-training paradigm for LLM reasoning. PPO (Schulman et al., 2017) served as the canonical optimizer in early RLHF work (Ouyang et al., 2022), but REINFORCE-style variants have since gained traction for lower complexity and favourable scaling (Williams, 1992; Arash et al., 2024; Kool et al., 2019a). GRPO (AI, 2024; 2025) normalizes rewards within each prompt group to estimate advantages without a critic, and DAPO (Yu et al., 2025) improves stability through dynamic sampling and entropy-preserving clipping. Recent analysis suggests that the reasoning capacity of RL with verifiable rewards remains bounded by the base model's distribution (Liu et al., 2025), and that data selection can matter as much as the optimizer (Xiong et al., 2025; Mao et al., 2026).

B Chat Templates and Prompt Formatting

ChatML System and User Template

```
<|im_start|>system
You are a helpful mathematical reasoning assistant.
<|im_end|>
<|im_start|>user
Please reason step by step, and
put your final answer within \boxed{}.
{Question}
<|im_end|>
<|im_start|>assistant
```

C Details of Models and Datasets

Models. We evaluate two families of open-weight Qwen reasoning models. The Qwen2.5-Math line, including Qwen2.5-Math-1.5B-Instruct (`https://huggingface.co/Qwen/Qwen2.5-Math-1.5B-Instruct`) and Qwen2.5-Math-7B-Instruct (`https://huggingface.co/Qwen/Qwen2.5-Math-7B-Instruct`), is specialized for mathematical reasoning and provides our main math-centric backbone. The Qwen3 line, including Qwen3-1.7B (`https://huggingface.co/Qwen/Qwen3-1.7B`), Qwen3-4B (`https://huggingface.co/Qwen/Qwen3-4B`), and Qwen3-8B (`https://huggingface.co/Qwen/Qwen3-8B`), offers broader reasoning capacity and stronger general-purpose post-training behavior; for all Qwen3 experiments we enable thinking mode throughout training and evaluation. According to the corresponding Hugging Face model cards, all five checkpoints are released under the Apache-2.0 license.

Datasets. We train on three public mathematical reasoning datasets with progressively increasing difficulty. GSM8K (`https://huggingface.co/datasets/openai/gsm8k`) contains relatively short grade-school arithmetic word problems and serves as the easiest setting in our study. MATH (`https://huggingface.co/datasets/EleutherAI/hendrycks_math`) contains substantially harder competition-style problems spanning algebra, geometry, counting, and number theory. DAPO-MATH (English split; `https://huggingface.co/datasets/BytedTsinghua-SIA/DAPO-Math-17k`) is the most challenging training source used here, with longer solution trajectories and more diverse reasoning patterns, making it especially suitable for probing efficiency gains under harder post-training regimes. For evaluation, we use the corresponding held-out test splits of GSM8K, MATH, and DAPO-MATH from the same repositories, together with the OOD benchmarks AMC23 (`https://huggingface.co/datasets/JingzeShi/amc23`) and AIME25 (`https://huggingface.co/datasets/math-ai/aime25`); these are exactly the benchmark sets reported in Table 1 and Table 2 of Section 5.

Licenses. The external assets used in our experiments are all taken from public model or dataset cards that state their licenses explicitly. The Qwen checkpoints are Apache-2.0; GSM8K and MATH are MIT; DAPO-MATH and AIME25 are Apache-2.0; and the AMC23 mirror used in our evaluation setup is MIT.

D Hardware and Runtime Details

All experiments are conducted on 8 NVIDIA Blackwell-series GPUs. For runs trained on GSM8K or MATH, each configuration requires roughly 3.5 hours on average. Runs trained on DAPO-MATH are longer because of the more difficult prompts and longer reasoning trajectories, with an average duration of about 8 hours per configuration. Relative to the full-data baseline, our method reduces end-to-end runtime by approximately 30%–40% across the reported settings.

E Detailed Computational Efficiency Analysis

In this section, we provide a rigorous mathematical derivation of the computational footprint associated with our reinforcement learning (RL) training pipeline. We contextualize our footprint within the standard `veRL` framework execution to derive the baseline theoretical coefficient of 10, which we partition into inference and optimization phases, measured in floating-point operations (FLOPs) per parameter per token.

Let P denote the number of parameters in the policy model, and T represent the average total sequence length ($T = L_{\text{prompt}} + L_{\text{response}}$).

Inference and Generation Cost. During the data collection phase, the policy model autoregressively generates n_{rollout} responses for each prompt. Generating a single token requires a forward pass through the active policy network. In dense causal Transformer architectures, KV-cache-assisted token decoding incurs approximately $2P$ FLOPs per token. Following standard engineering implementations of GRPO (e.g., the `veRL` framework using vLLM), the evaluation for the reference model logic is computationally merged or evaluated without gradient tracking. A reward model is subsequently invoked to evaluate the sequences, costing an additional $2P$ FLOPs per token for the forward pass. Therefore, the generative inference scaling factor analytically resolves to: Eq. (9):

$$C_{\text{infer}} = \underbrace{2}_{\text{Actor Generation}} + \underbrace{2}_{\text{Reward Model Scoring}} = 4. \tag{9}$$

Optimization and Update Cost. During the optimization phase, only the actor (policy) model's weights undergo iterative updates via backpropagation. No critic model is utilized in GRPO. For a decoder-only LLM, the optimization phase incorporates a forward pass (activations) and a backward pass (gradients with respect to activations and learnable weights). The forward pass necessitates $2P$ FLOPs per token. The backward pass demands approximately $4P$ FLOPs per token ($2P$ for activation gradients, and $2P$ for weight gradients). The underlying computational coefficient for optimization is modeled strictly as: Eq. (10):

$$C_{\text{optim}} = \underbrace{2}_{\text{Training Forward}} + \underbrace{4}_{\text{Training Backward}} = 6. \tag{10}$$

Total Baseline and Data Selection Savings. Cumulatively, the theoretical cost baseline per token scales with $C_{\text{total}} = C_{\text{infer}} + C_{\text{optim}} = 10$. For a global dataset D and epoch count E, the baseline expense is given by Eq. (11):

$$\text{FLOPs}_{\text{base}} \approx 10 \cdot E \cdot |D| \cdot n_{\text{rollout}} \cdot P \cdot T. \tag{11}$$

Our learning-zone data filtering regimen dynamically selects and retains only the prime 40% of sample groups mapping backpropagation. Generative inferences and objective energy scorings remain necessarily fully evaluated. The modified compute footprint explicitly rescales the optimization term, yielding Eq. (12):

$$\text{FLOPs}_{\text{ours}} \approx (C_{\text{infer}} + 0.4 \cdot C_{\text{optim}}) \cdot E \cdot |D| \cdot n_{\text{rollout}} \cdot P \cdot T = (4 + 0.4 \cdot 6) \cdot (\dots) = 6.4 \cdot (\dots). \tag{12}$$

Consequently, the structural computation conserved corresponds firmly to Eq. (13):

$$\text{Saved FLOPs} = 1 - \frac{6.4}{10} = 36\%. \tag{13}$$

This estimate accounts for backward selection with full rollout coverage. The optional forward pruner reduces rollout generation separately, and its additional savings are measured in the wall-clock experiments.

F Additional Ablation Studies

F.1 Gumbel Perturbation

Controlled exploration. Finite-rollout pass-rate estimates can give prompt groups identical or nearly identical Energy Scores, making deterministic Top-K selection repeatedly favor the same groups. We perturb scores independently before ranking:

$$\tilde{E}_i = E_i^{(t)} + \tau\xi_i, \qquad \xi_i \overset{\text{i.i.d.}}{\sim} \text{Gumbel}(0,1), \qquad \tau = 0.05. \tag{14}$$

Here, the Gumbel parameters specify location and scale; equivalently, $\tau\xi_i \sim \text{Gumbel}(0, 0.05)$. For fixed scores and $\tau > 0$, the Gumbel-Max identity gives Top-1 selection probabilities

$$\Pr\left(i = \arg\max_j \tilde{E}_j\right) = \frac{\exp(E_i^{(t)}/\tau)}{\sum_{j\in\mathcal{A}} \exp(E_j^{(t)}/\tau)}. \tag{15}$$

Gumbel-Top-K extends this to sampling without replacement (Kool et al., 2019b); these Top-1 probabilities are not the marginal inclusion probabilities for $K > 1$. For two prompts with energy gap $\Delta = E_i^{(t)} - E_j^{(t)} > 0$, the probability of reversing their order is

$$\Pr(\tilde{E}_j > \tilde{E}_i) = \frac{1}{1 + \exp(\Delta/\tau)}. \tag{16}$$

Thus, perturbations randomize ties and allow exploration among nearby scores, while pairwise reversals become less likely as the gap grows relative to τ. The stochastic rule can also revisit groups whose unperturbed energy is zero.

Ablation results. We compare the default $\tau = 0.05$ with deterministic Top-K ($\tau = 0$) for three backbones. Table 3 reports the signed relative change $100(S_{0.05}/S_0 - 1)\%$, where S_τ is the evaluation score with perturbation scale τ and OOD Avg is the arithmetic mean of AMC23 and AIME25 Pass@1.

Table 3: Effect of Gumbel perturbation relative to deterministic Top-K. Values are relative percentage changes, not percentage-point differences.

Backbone	MATH change	OOD Avg change
Qwen2.5-Math-1.5B	$+0.3\%$	$+2.1\%$
Qwen2.5-Math-7B	-0.3%	$+13.7\%$
Qwen3-4B	-1.6%	$+17.3\%$

The perturbation improves OOD Avg in all three settings, with MATH changes ranging from -1.6% to $+0.3\%$. Its empirical benefit is concentrated in OOD transfer.

F.2 Interaction between the Difficulty Anchor and Momentum

The anchor captures difficulty under the initial policy, while momentum tracks changes relative to recent performance. With the default $\alpha = 0.3$, their interaction satisfies

$$0 \le E_i^{(t)} \le 1.3 D_i^{(0)}, \qquad m_i^{(t)} \le 1 - \mu_i^{(t-1)}. \tag{17}$$

Momentum therefore provides bounded modulation of the anchor-weighted uncertainty. Prompts that are initially easy and remain close to saturation have little room for further positive momentum. The component ablations in Figure 3a show that retaining both factors improves OOD performance on average.

Repeated-regression diagnostic. We also measure the fraction of MATH training prompts whose empirical accuracy decreases over two consecutive epoch transitions at least once during training (Table 4). The diagnostic covers the full training pool, including initially medium- and high-difficulty prompts. Repeated declines affect roughly one in ten training prompts on both backbones.

Table 4: Fraction of MATH training prompts exhibiting two consecutive epoch-level accuracy drops at least once. Values are mean ± standard deviation, with standard deviations expressed in percentage points.

Model	Fraction of training prompts
Qwen2.5-Math-1.5B	$(11.5 \pm 3.6)\%$
Qwen2.5-Math-7B	$(9.2 \pm 2.1)\%$

F.3 Comparison with Recent Online Selection Methods

We extend the comparisons to AdaRFT-PPO, DOTS, DOTS+RR, and DPS on MATH, following the configurations in their original papers. AdaRFT-PPO (Shi et al., 2025) adapts the target difficulty of sampled problems using recent reward signals. DOTS (Sun et al., 2025) uses learned difficulty prediction to select prompts before rollout, and DOTS+RR additionally reuses recent rollouts; LZE instead ranks observed prompt groups for policy updates using a closed-form score computed from rollout statistics. DPS (Mao et al., 2026) uses a hidden Markov model to predict prompt-solving dynamics and select prompts before rollout. Table 5 reports performance relative to Vanilla-RL, the full-data baseline without data selection.

Table 5: Additional method comparisons on MATH-trained models. Entries are signed relative gains $100(s_{\text{method}}/s_{\text{Vanilla-RL}} - 1)\%$. OOD Avg is the arithmetic mean of AMC23 and AIME25 Pass@1.

Method	Qwen2.5-Math-1.5B		Qwen2.5-Math-7B	
	MATH	OOD Avg	MATH	OOD Avg
Vanilla-RL	0.0%	0.0%	0.0%	0.0%
AdaRFT-PPO	$+1.7\%$	$+13.3\%$	$+3.9\%$	-12.0%
DOTS	-1.4%	$+33.2\%$	$+1.8\%$	-10.3%
DOTS+RR	-0.6%	$+29.4\%$	$+1.8\%$	-6.9%
DPS	-1.2%	$+18.2\%$	$+3.0\%$	-15.3%
LZE	$+0.0\%$	$+23.2\%$	$+1.1\%$	$+5.9\%$

LZE improves OOD Avg over Vanilla-RL on both evaluated backbones while maintaining comparable MATH performance.

Training speed. Table 6 compares policy-training-loop speed on the same machine. We define speed as $v = 1/t_{\text{step}}$, where t_{step} is the mean wall-clock time per training step, and report $100(v_{\text{LZE}}/v_{\text{method}} - 1)\%$. LZE increases training speed relative to both DOTS and DOTS+RR on the two backbones.

Table 6: Relative training-speed gains of LZE over DOTS and DOTS+RR on MATH.

Backbone	vs. DOTS	vs. DOTS+RR
Qwen2.5-Math-1.5B	$+73.7\%$	$+59.4\%$
Qwen2.5-Math-7B	$+60.8\%$	$+42.3\%$

G Additional Analysis

Centered binary rewards. For a group with empirical pass rate $\hat{p} = n^{-1}\sum_{k=1}^{n} r_k$, using $r_k^2 = r_k$ gives

$$\frac{1}{n}\sum_{k=1}^{n}(r_k - \hat{p})^2 = \frac{1}{n}\sum_{k=1}^{n} r_k^2 - \hat{p}^2 = \hat{p}(1-\hat{p}). \tag{18}$$

When $r_k \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(p)$,

$$\mathbb{E}[\hat{p}(1-\hat{p})] = \frac{n-1}{n}p(1-p), \qquad \Pr(0 < \hat{p} < 1) = 1 - p^n - (1-p)^n. \tag{19}$$

For $n \geq 2$, both quantities are maximized at $p = 1/2$ and vanish at the endpoints.

G.1 Score-function second moment

Fix a prompt x_i and the current policy, and write $p_i = \mathbb{E}[r(x_i, y)]$ and $s_i(y) = \nabla_\theta \log \pi_\theta(y \mid x_i)$. Assume finite second moments and the usual regularity conditions for differentiating expectations, so that $\mathbb{E}[s_i] = 0$ and $\nabla_\theta p_i = \mathbb{E}[r s_i]$. For $0 < p_i < 1$, define the reward-conditioned squared score norms

$$G_{i,b} = \mathbb{E}\big[\|s_i(y)\|^2 \mid r(x_i, y) = b\big], \qquad b \in \{0, 1\}. \tag{20}$$

Theorem 1 (Second moment with a population baseline)**.** *Let $z_i = (r - p_i)s_i(y)$ and let $\bar{z}_i = n^{-1}\sum_{k=1}^{n} z_{i,k}$ for n independent rollouts from the current policy. Then*

$$Q_i := \mathbb{E}\|z_i\|^2 = p_i(1-p_i)\big[(1-p_i)G_{i,1} + p_i G_{i,0}\big], \tag{21}$$

$$\operatorname{tr}\operatorname{Cov}(\bar{z}_i) = \frac{Q_i - \|\nabla_\theta p_i\|^2}{n}. \tag{22}$$

Proof. Conditioning on the binary reward gives

$$\mathbb{E}\|z_i\|^2 = p_i(1-p_i)^2 G_{i,1} + (1-p_i)p_i^2 G_{i,0} \tag{23}$$

$$= p_i(1-p_i)\big[(1-p_i)G_{i,1} + p_i G_{i,0}\big]. \tag{24}$$

The score identity implies $\mathbb{E}[z_i] = \mathbb{E}[r s_i] - p_i\mathbb{E}[s_i] = \nabla_\theta p_i$. Independence of the rollouts then yields

$$\operatorname{tr}\operatorname{Cov}(\bar{z}_i) = \frac{1}{n}\left(\mathbb{E}\|z_i\|^2 - \|\mathbb{E}[z_i]\|^2\right). \tag{25}$$

□

When $G_{i,1} \approx G_{i,0} \approx G_i$, Eq. (21) reduces to $Q_i \approx p_i(1-p_i)G_i$. The exact decomposition retains differences in score norms between successful and unsuccessful trajectories; the variance additionally depends on the squared mean gradient. For the empirical group baseline used in Eq. (2), the centered terms share $\hat{p}$. Their group gradient $g_i = n^{-1}\sum_k (r_k - \hat{p})s_i(y_k)$ has expectation

$$\mathbb{E}[g_i] = \left(1 - \frac{1}{n}\right)\nabla_\theta p_i, \tag{26}$$

since $\mathbb{E}[\hat{p}s_i(y_k)] = n^{-1}\nabla_\theta p_i$. The centered-reward identities above apply directly to this empirical baseline.

G.2 Finite-time EMA and momentum

Starting from $\mu_i^{(0)} = p_i^{(0)}$, unrolling the EMA gives

$$\mu_i^{(t)} = \lambda^t p_i^{(0)} + (1-\lambda)\sum_{s=1}^{t} \lambda^{t-s} p_i^{(s)}. \tag{27}$$

For $t \geq 2$, substituting the EMA recurrence into the definition of momentum yields

$$m_i^{(t)} = p_i^{(t)} - \lambda \mu_i^{(t-2)} - (1-\lambda) p_i^{(t-1)} \tag{28}$$

$$= \Delta p_i^{(t)} + \lambda m_i^{(t-1)}. \tag{29}$$

Since $m_i^{(1)} = p_i^{(1)} - p_i^{(0)} = \Delta p_i^{(1)}$, induction gives Eq. (8). Thus, momentum measures signed changes accumulated over recent observations, with older changes attenuated by powers of λ. For the centered input $q_i^{(t)} = p_i^{(t)} - p_i^{(0)}$, the transfer function to momentum is $H(z) = (1-z^{-1})/(1-\lambda z^{-1})$, with zero gain at zero frequency, giving the causal high-pass interpretation.