ON THE DIRECTION OF RLVR UPDATES FOR LLM REASONING: IDENTIFICATION AND EXPLOITATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards (RLVR) has substantially improved the reasoning capabilities of large language models. While existing analyses identify that RLVR-induced changes are sparse, they primarily focus on the **magnitude** of these updates, largely overlooking their **direction**. In this work, we argue that the direction of updates is a more critical lens for understanding RLVR's effects, which can be captured by the signed, token-level log probability difference $\Delta \log p$ between the base and final RLVR models. Through statistical analysis and token-replacement interventions, we demonstrate that $\Delta \log p$ more effectively identifies sparse, yet reasoning-critical updates than magnitude-based metrics (*e.g.,* divergence or entropy). Building on this insight, we propose two practical applications: (1) a *test-time extrapolation* method that amplifies the policy along the learned $\Delta \log p$ direction to improve reasoning accuracy without further training; (2) a *training-time reweighting* method that focuses learning on low-probability (corresponding to higher $\Delta \log p$) tokens, which improves reasoning performance across models and benchmarks. Our work establishes the direction of change as a key principle for analyzing and improving RLVR.

1 INTRODUCTION

Recent advances have substantially improved the reasoning capabilities of large language models, giving rise to powerful reasoning-centric models such as OpenAI o1 (Jaech et al., 2024), Deepseek R1 (Guo et al., 2025), Gemini 2.5 (Comanici et al., 2025), and Qwen3 (Yang et al., 2025a). A key algorithmic driver of this progress is reinforcement learning with verifiable rewards (RLVR) (Guo et al., 2025; Team, 2025; Yang et al., 2025a), which fine-tunes a model's generation policy using feedback from task-specific verifiers, thereby eliciting and amplifying the reasoning ability.

To elucidate how RLVR confers its gains, a natural lens is to compare what changes in the final RL-trained model π_{RL} relative to its base counterpart π_{Base} (Ren & Sutherland, 2025). Previous analyses have consistently shown that the RLVR-induced changes are sparse, impacting only a small subset of tokens in the output sequence. For example, Wang et al. (2025b) associate these changes with high-entropy tokens, Huan et al. (2025) corroborate the sparsity by measuring the KL divergence between π_{Base} and π_{RL}, while Yang et al. (2025b) and Deng et al. (2025) attribute this sparsity to selective gradient updates during RLVR training. However, these studies primarily emphasize the **magnitude of change** while largely overlooking its **direction** in the model's output distribution. As shown in Fig. 1(b), magnitude-based metrics (*e.g.,* entropy, KL divergence) yield nearly identical histograms for the base and final RLVR models, indicating that magnitude alone is insufficient to characterize the transformation from π_{Base} to π_{RL}.

To address this gap, we directly quantify directional shifts in the model's distribution using the signed, token-level log-probability difference:

$$\Delta \log p(y_t|x, y_{<t}) = \log \pi_{\mathrm{RL}}(y_t|x, y_{<t}) - \log \pi_{\mathrm{Base}}(y_t|x, y_{<t}), \tag{1}$$

which captures how RLVR shifts the probability mass on each token, with positive values indicating increased probabilities and negative values vice versa. As shown in Fig. 1(b), histograms of $\Delta \log p$ exhibit a clear bimodal pattern with two distinct tails, highlighting a clear directional signature absent in magnitude-based metrics. This metric can reveal which token RLVR prioritizes, such as reasoning-critical tokens (*e.g.,* those enhancing reasoning correctness) versus irrelevant ones. We

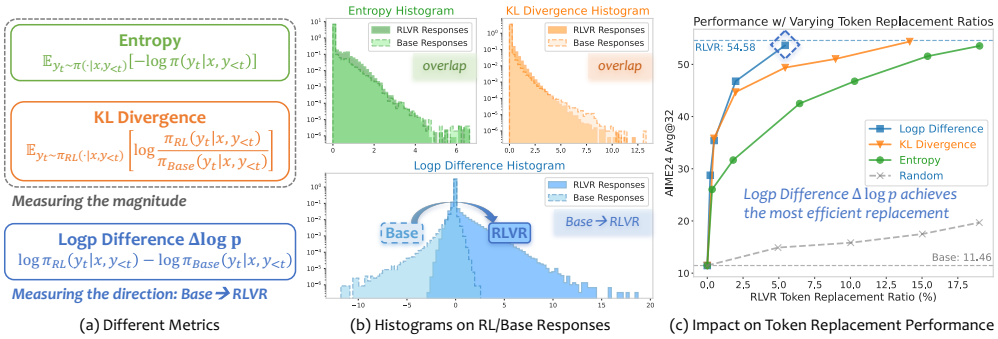


Figure 1: (a) Token-level metrics for analyzing RLVR updates. (b) Histograms of each metric on responses generated by base and RLVR models. With a log-scale y-axis, most values concentrate near zero for all metrics, but only $\Delta \log p$ shows a directional shift distinguishing RLVR from the base model. (c) Token-replacement performance: replacing base tokens with RLVR choices at positions selected by each metric, where $\Delta \log p$ recovers RLVR performance with the fewest replacements.

further validate its utility via a token replacement intervention (Deng et al., 2025): for each metric, we identify salient positions and replace the base model's tokens with the RLVR model's choices at those positions during generation. As shown in Fig. 1(c), selecting by $\Delta \log p$ reaches RLVR-level performance with the fewest substitutions, pinpointing tokens where RLVR learns reasoning-critical updates rather than incidental deviations. These findings underscore a key principle: *analyzing the direction of changes, rather than solely their magnitude, provides deeper insights.* The signed log-probability difference provides a practical and effective handle for this diagnostic analysis.

Building on this principle, we first propose a test-time augmentation that extrapolates the RLVR policy's distribution along the $\Delta \log p$ direction for reasoning-critical tokens selectively, amplifying reasoning-related updates and improving accuracy without additional training. Furthermore, we observe that tokens with the largest $\Delta \log p$ consistently correspond to low-probability tokens during RLVR training. Motivated by this, we design a probability-aware reweighting of policy-gradient advantages, upweighting contributions from low-probability tokens to focus learning on reasoning-critical positions as $\Delta \log p$ indicated. This reweighting yields additional gains over current state-of-the-art RLVR methods (*e.g.,* DAPO (Yu et al., 2025)) across diverse benchmarks and models.

In summary, this work introduces a directional diagnostic for analyzing RLVR's effects and, based on these findings, develops two practical strategies for reasoning enhancement: a test-time extrapolation technique and a training-time reweighting method. We hope our work offers a new perspective for analyzing and improving RLVR through the lens of update direction.

2 Preliminaries

Group Relative Policy Optimization (GRPO). GRPO (Shao et al., 2024) is a variant of the milestone policy gradient algorithm PPO (Schulman et al., 2017). It is adapted for LLM training by eliminating the need for a separate critic model. For each QA pair (x, a) sampled from dataset $\mathcal{D}$, GRPO generates a group of G responses $\{y_i\}_{i=1}^{G}$ using the old policy $\pi_{\theta_{\text{old}}}$, computes their rewards $\{R_i\}_{i=1}^{G}$, and estimates the advantage of each response in a group-relative manner:

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_i\}_{i=1}^{G})}{\text{std}(\{R_i\}_{i=1}^{G})}. \tag{2}$$

Then the policy π_θ is optimized by maximizing the following objective:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\theta) =& \mathbb{E}_{(x,a)\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\\ & \text{clip}\big(r_{i,t}(\theta),1-\epsilon,1+\epsilon\big)\,\hat{A}_{i,t}\Big) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})\Bigg],\end{aligned} \tag{3}$$

where $r_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t}|x,y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x,y_{i,<t})}$ is the importance sampling ratio, ϵ is the clipping range for $r_{i,t}(\theta)$, and $\mathbb{D}_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})$ regularizes the policy to stay close to a reference policy π_{ref}.

Dynamic Sampling Policy Optimization (DAPO). DAPO(Yu et al., 2025) is a state-of-the-art critic-free RLVR algorithm that further refines GRPO. It introduces several techniques, including clip-higher mechanism, dynamic sampling strategy, token-level loss aggregation, overlong punishment, and removing the KL penalty. DAPO's objective is defined as:

$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{(x,a)\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\left[\frac{1}{\sum_{i=1}^{G}|y_i|}\sum_{i=1}^{G}\sum_{t=1}^{|y_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\right.$$
$$\left.\text{clip}\big(r_{i,t}(\theta), 1-\epsilon_{\text{low}}, 1+\epsilon_{\text{high}}\big)\hat{A}_{i,t}\Big)\right], \quad s.t.\ 0 < |\{y_i \mid \text{is\_equivalent}(a, y_i)\}| < G. \tag{4}$$

Given its success, we adopt DAPO as the primary baseline algorithm for our empirical analysis.

Token-level metrics for RLVR analysis. To study how RLVR turns a base model into the RL-finetuned counterpart, we mainly compare the following token-level metrics for RLVR analysis:

- Entropy: Wang et al. (2025b) observed that RLVR-induced changes are sparse and tend to concentrate on high-entropy tokens. This token-level entropy is defined as:

$$\mathcal{H}_\pi(y_t|x, y_{<t}) = \mathbb{E}_{y_t\sim\pi(\cdot|x,y_{<t})}[-\log\pi(y_t|x, y_{<t})]. \tag{5}$$

  We calculate this entropy for both the RLVR model $\mathcal{H}_{\pi_{\text{RL}}}$ and the base model $\mathcal{H}_{\pi_{\text{Base}}}$.

- Divergences: Huan et al. (2025) used KL Divergence to quantify the distributional shift, also finding that the changes are sparse. The token-level KL divergence is defined as:

$$\mathbb{D}^{\text{KL}}_{\pi_{\text{RL}},\pi_{\text{Base}}}(y_t|x, y_{<t}) = \mathbb{E}_{y_t\sim\pi_{\text{RL}}(\cdot|x,y_{<t})}\left[\log\frac{\pi_{\text{RL}}(y_t|x, y_{<t})}{\pi_{\text{Base}}(y_t|x, y_{<t})}\right]. \tag{6}$$

  We also include its reversed variant $\mathbb{D}^{\text{KL}}_{\pi_{\text{Base}},\pi_{\text{RL}}}$ and the averaged KL Divergence $\mathbb{D}^{\text{KL}} = \frac{1}{2}(\mathbb{D}^{\text{KL}}_{\pi_{\text{RL}},\pi_{\text{Base}}} + \mathbb{D}^{\text{KL}}_{\pi_{\text{Base}},\pi_{\text{RL}}})$ to avoid asymmetry bias for a comprehensive analysis.

3 Dissecting the Token-Level Changes Introduced by RLVR

This section aims to dissect the token-level mechanisms through which RLVR training transforms a base model into its fine-tuned counterpart. First, we show that the logp difference ($\Delta\log p$, Eq. 1) captures directional shifts in probability mass and separates base from RLVR generations, whereas magnitude-only metrics (entropy/divergence) do not. Second, we conduct a token replacement experiment to validate that $\Delta\log p$ more precisely identifies sparse, reasoning-critical tokens targeted by RLVR. Finally, we explain the sparsity through a gradient analysis showing that policy updates concentrate on low-probability tokens of RLVR's policy gradient updates.

3.1 Statistical Analysis: Directional vs. Magnitude-Based Metrics

Experimental Setup. We conduct a statistical analysis on outputs from several RLVR-base model pairs (ORZ (Hu et al., 2025a), DAPO (Yu et al., 2025), UniReason (Huan et al., 2025)) to compare how different token-level metrics capture RLVR-induced changes. We plot histograms of entropy, divergences, and logp difference of different models' generated tokens on the AIME-24 dataset.

Statistical Comparison. Fig. 1(b) shows the distributions of these metrics for the UniReason model pair. Across all metrics, the histograms are sharply peaked near zero (note the log-scale y-axis), confirming that RLVR-induced changes are sparse.[1] However, the entropy and KL divergence distributions are nearly identical for both the base and RLVR model outputs. In contrast, the $\Delta\log p$ distribution exhibits two distinct tails: a positive tail corresponding to tokens favored by the RLVR model and a negative tail for the base model. This pattern holds across all tested model pairs and for multiple entropy/divergence variants (Appx. D): the distributions of magnitude-based metrics are nearly indistinguishable between tokens generated by the RLVR and base models (Figs. 9-11), whereas $\Delta\log p$ consistently exhibits clear bimodal patterns (Fig. 8).

[1] Wang et al. (2025b) argue that RLVR primarily modifies tokens with high entropy. The observed concentration of near-zero-entropy tokens is therefore consistent with sparse updates under their assumptions.

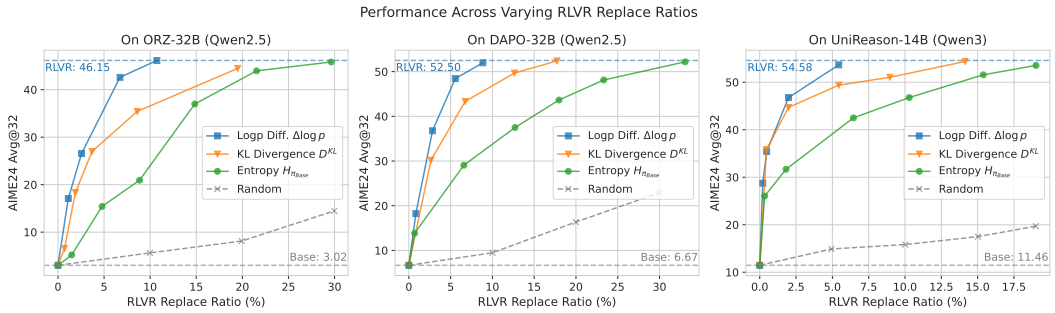


Figure 2: Token-replacement performance across metrics and model pairs. While all metrics can recover RLVR-level accuracy, $\Delta \log p$ does so with *the fewest replacements*, demonstrating its precision in isolating the reasoning-critical minor tokens changed by RLVR training.

This is because magnitude-only metrics quantify the size of the distributional change but **ignore its direction,** *i.e.,* whether a given token is more favored by the RLVR model or the base model. With directional information, $\Delta \log p$ **reveals a clear difference between the two modes**, enabling more precise identification of the sparse, reasoning-enhancing updates induced by RLVR, and we will validate their impact on reasoning performance in the following section.

3.2 Recovering RLVR Performance via Selective Token Replacement

Token Replacement Setup. To further assess how the minority tokens identified by each metric affect reasoning ability, we conduct a *selective token replacement* experiment adapted from Deng et al. (2025). At each decoding step, we sample a token from π_{Base}, then apply a metric-specific criterion f^τ to decide whether to replace the token with one sampled from π_{RL} (Alg. 1). The threshold τ is adjusted to control replacement rates across metrics, enabling fair comparisons.

Algorithm 1 Selective Token Replacement

Require: Base and RLVR models $\pi_{\text{Base}}, \pi_{\text{RL}}$, prompt x, criterion function $f^\tau(\cdot) \in \{0, 1\}$

Initialize response: $t \leftarrow 0, y_{\leq 0} \leftarrow$ ""
while $y_t \neq$ "<EOS>" **do**
  $t \leftarrow t + 1$
  Sample from base: $y_t \sim \pi_{\text{Base}}(\cdot|x, y_{<t})$
  if $f^\tau(y_t|x, y_{<t}) = 1$ **then**
    Replace the token: $y_t \sim \pi_{\text{RL}}(\cdot|x, y_{<t})$
  end if
end while
return $y_{\leq t}$

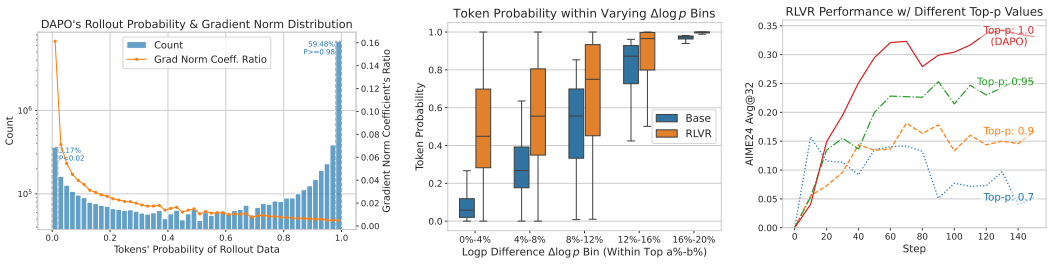
We compare entropy, KL Divergences[2], and logp difference, with the corresponding replacement criteria functions defined as follows:

- Entropy: Following the hypothesis that RLVR updates target high-entropy positions (Wang et al., 2025b), we replace the base model's token if its token distribution has entropy exceeding a threshold τ: $f^\tau_{\mathcal{H}}(y_t|x, y_{<t}) = \mathbb{I}(\mathcal{H}(y_t|x, y_{<t}) > \tau)$.
- KL Divergences: Similarly, to target positions where the two models diverges most, we replace the token if the divergence is greater than τ: $f^\tau_{\mathbb{D}}(y_t|x, y_{<t}) = \mathbb{I}\big(\mathbb{D}(y_t|x, y_{<t}) > \tau\big)$.
- Logp Difference: A large negative $\Delta \log p$ for a token y_t indicates that RLVR has learned to penalize it relative to the base model. We exploit this by replacing tokens whose logp difference falls *below* a threshold τ: $f^\tau_{\text{logp}}(y_t|x, y_{<t}) = \mathbb{I}\big(\Delta \log p(y_t|x, y_{<t}) < \tau\big)$.

This selective replacement setup, controlled by the metric-specific thresholds, allows us to compare the impact of tokens identified by each metric on reasoning performance at matched replacement rates. Fig. 2 shows results on AIME-24 for three representative metrics $\mathcal{H}_{\pi_{\text{Base}}}$, $\mathbb{D}^{\text{KL}}$, and $\Delta \log p$, while Fig. 6 in Appx. A.2 provides ablations with additional metrics, including the RLVR model's entropy $\mathcal{H}_{\pi_{\text{RL}}}$ and KL-divergence variants. All metrics are contrasted with a random baseline that uniformly replaces tokens: $f^\tau_{\text{rand}}(\cdot) = \mathbb{I}_{\rho \sim U[0,1]}(\rho < \tau)$. The key observations are as follows:

Observation I: Selectively replacing a minority of base models' tokens can recover RLVR performance. As shown in Fig. 2, replacing 5-30% of a base model's sampled tokens with different

[2]We mainly use the averaged KL divergence $\mathbb{D}^{\text{KL}} = \frac{1}{2}(\mathbb{D}^{\text{KL}}_{\pi_{\text{RL}}, \pi_{\text{Base}}} + \mathbb{D}^{\text{KL}}_{\pi_{\text{Base}}, \pi_{\text{RL}}})$ for token replacement to avoid potential asymmetry bias and include KL's variants $\mathbb{D}^{\text{KL}}_{\pi_{\text{RL}}, \pi_{\text{Base}}}$ and $\mathbb{D}^{\text{KL}}_{\pi_{\text{Base}}, \pi_{\text{RL}}}$ for ablation study.



(a) Gradient norm and probability (b) Token probability *v.s.* $\Delta \log p$ (c) RLVR performance *v.s.* top-p

Figure 3: (a) Token probability and gradient norm coefficient $1 - \pi_\theta(\cdot)$ of a DAPO step, where the gradient concentrates on rare, low-probability tokens. (b) Token probability within different $\Delta \log p$ bins, where higher $\Delta \log p$ bins contain lower probability for both base and RLVR models. (c) Effect of top-p filtering on RLVR training performance. Performance declines with more filtering.

metrics suffices to match the final RLVR model's accuracy. In contrast, randomly replacing the tokens without metric selection produces much slower performance growth. This demonstrates that RLVR-modified tokens are sparsely distributed along the sequence but disproportionately important for reasoning, highlighting the efficacy of the evaluated metrics in identifying these critical tokens.

Observation II: Logp difference > divergence > entropy in identifying RLVR-learned reasoning patterns. Across all model pairs (Fig. 2), $\Delta \log p$-based replacement reaches the RLVR model's accuracy with the *fewest* substitutions (around *10%* of tokens). In comparison, magnitude-only metrics (*e.g.,* divergence and entropy) require clearly more replacement to match RLVR performance, indicating lower precision in identifying reasoning-critical changes introduced by RLVR. Between these two, divergence consistently outperforms entropy, suggesting that RLVR changes may not be restricted to high-entropy positions. This ordering—$\Delta \log p$ highest, followed by divergence, then entropy—remains stable across different divergence and entropy variants (Fig. 6 in Appx. A.2), further validating the superiority of logp difference in isolating the most influential positions.

3.3 A Gradient-Based Explanation for the Sparse Updates

Our previous analysis established that the RLVR model differs from its base counterpart on a small but critical subset of tokens most effectively identified by $\Delta \log p$. Here, we provide a gradient-based explanation for this sparsity of changes: RLVR's policy gradient inherently concentrates updates on rare, low-probability tokens, correlating with tokens with high $\Delta \log p$ in the final model.

RLVR's policy gradient sparsely concentrates on low-probability tokens. The gradient of the DAPO objective $\mathcal{J}_{\text{DAPO}}$ for an un-clipped token $y_{i,t}$ can be written as $w_{i,t} \cdot \nabla_\theta \log \pi_\theta(y_{i,t}|x, y_{i,<t})$, where $w_{i,t} = r_{i,t}(\theta)\hat{A}_{i,t}$ combines the importance sampling ratio and advantage. To analyze the token's gradient norm, we have the following lemma (see the proof in Appx. C):

Lemma 3.1. *For a softmax-parameterized LLM policy with logits vector z for the output token $y_{i,t}$, the $\ell 1$-norm of the DAPO objective's gradient w.r.t. z is given by:*

$$\|\nabla_z \mathcal{J}_{\text{DAPO}}(y_{i,t}|x, y_{i,<t})\|_1 = 2|w_{i,t}| \cdot \big(1 - \pi_\theta(y_{i,t}|x, y_{i,<t})\big).$$

This partial gradient's $\ell 1$-norm directly depends on $1 - \pi_\theta(y_i|x, y_{i,<t})$, with larger gradient sizes for lower-probability tokens. Furthermore, Yang et al. (2025b) formally proved that the full gradient norm is tightly bounded by the $1 - \pi_\theta(\cdot)$ term. Consequently, low-probability tokens, despite their rarity, receive disproportionately large gradient updates. We corroborate this empirically in Fig. 3(a), which plots tokens' probability and their gradient coefficient from an intermediate DAPO training step. Although low-probability tokens are sampled infrequently, they account for most of the total gradient mass. This concentration of gradients explains why RLVR's modifications are sparse: learning is naturally focused on a small, high-impact set of low-probability positions.

High $\Delta \log p$ tokens are the updated low-probability tokens. To complete the argument, we link the low-probability tokens that dominate training updates to the high-$\Delta \log p$ tokens in the final model. Fig. 3(b) analyzes tokens grouped by their $\Delta \log p$ values. It reveals two patterns: first, the

probability of tokens in high-$\Delta \log p$ bins increases substantially from the base to the RLVR model; second, these high-$\Delta \log p$ tokens have clearly lower probabilities in both models. This confirms that the most significant updates learned by RLVR target those low-probability tokens, and the sparsity of RLVR's changes is therefore a direct consequence of sparse, high-magnitude gradients acting on these critical tokens, which can be effectively identified post-hoc by their large $\Delta \log p$.

Excluding low-probability tokens during training impairs performance. To causally verify the importance of these low-probability tokens, we conduct a training-time intervention experiment to provide direct evidence for our hypothesis. We train the Qwen2.5-Math-7B base model (Yang et al., 2024) using DAPO but adopt a top-p sampling strategy during rollout to filter out low-probability tokens. The results, plotted in Fig. 3(c), are conclusive. Even a mild filter (*e.g.,* top-p=0.95) leads to a substantial drop in performance compared to the default setting (top-p=1.0). As the filter becomes more aggressive (*i.e.,* with lower top-p thresholds), performance degrades sharply. This experiment demonstrates that these low-probability tokens are not merely correlated with gradient size but are essential for the reasoning improvements achieved by RLVR training.

Takeaway

1. **RLVR's gains stem from sparse, high-impact modifications.** Our analysis reveals that RLVR's performance gains originate not from a global distribution shift, but from targeted, high-impact changes to a minority of tokens.
2. **Logp difference pinpoints these sparse changes.** By capturing the direction of probability shifts from base to RLVR, logp difference outperforms magnitude-only metrics like entropy or divergence in isolating the reasoning-critical tokens that RLVR learns.
3. **Sparsity originates from RLVR's focus on low-probability tokens.** The sparse difference is explained by the inherent concentration of RLVR's gradients on rare, low-probability tokens, making these tokens the focal point for improvement and the source of the sparse, high-$\Delta \log p$ changes we observe.

4 Exploiting RLVR's Directional Updates to Boost Reasoning

Building on Sec. 3, which isolates sparse and directional updates via $\Delta \log p$, we propose two practical strategies to utilize this directional learning: (i) a *test-time selective extrapolation* that shifts probability mass further along the learned direction on critical tokens; (ii) a *training-time advantage reweighting* that prioritizes low-probability tokens implicated by high $\Delta \log p$. Both methods provide practical ways to boost performance by exploiting the directional mechanisms of RLVR.

4.1 Test-Time Enhancement via Extrapolation

Selective test-time extrapolation along the $\Delta \log p$ direction. Our token replacement experiment demonstrated that $\Delta \log p$ effectively identifies the reasoning-critical changes of RLVR. This raises a natural question: Can we move beyond simple replacement and actively amplify these critical changes to surpass the RLVR model's performance? We therefore instantiate a token-level extrapolation: treat $\Delta \log p = \log \pi_{\text{RL}}(\cdot) - \log \pi_{\text{Base}}(\cdot)$ as a learned "reasoning direction" pointing from base to RLVR distribution. Our strategy is to amplify this signal by extrapolating the RLVR model's distribution further along this direction. The extrapolated policy $\pi^{\gamma}_{\text{Extra}}$ is given by:

$$\begin{aligned} \log \pi^{\gamma}_{\text{Extra}}(y_t|x, y_{<t}) &:= \log \pi_{\text{RL}}(y_t|x, y_{<t}) + \gamma \cdot \Delta \log p(y_t|x, y_{<t}) + z(x, y_{<t}) \\ &= (1+\gamma) \cdot \log \pi_{\text{RL}}(y_t|x, y_{<t}) - \gamma \cdot \log \pi_{\text{Base}}(y_t|x, y_{<t}) + z(x, y_{<t}), \end{aligned} \tag{7}$$

where γ is a hyperparameter controlling the extrapolating strength, and $z(\cdot)$ is a log-partition function. In probability space, this is equivalent to re-weighting the RLVR distribution:

$$\pi^{\gamma}_{\text{Extra}}(y_t|x, y_{<t}) \propto \pi_{\text{RL}}(y_t|x, y_{<t}) \cdot \exp\big(\gamma\, \Delta \log p(y_t|x, y_{<t})\big).$$

This framing connects our method to reward-guided decoding literature (Khanov et al., 2024; Liu et al., 2024; Xu et al., 2025), where a reward function is used to re-weight the probability distribution. Our $\Delta \log p$ thereby acts as a token-level reward that encourages better reasoning in this framework.

Why selective? RLVR's improvements concentrate on a minority of tokens; most positions exhibit negligible $\Delta \log p$. A global intervention risks distorting well-calibrated tokens. We therefore apply extrapolation *selectively*, using f^{τ}_{logp} to gate positions with large negative $\Delta \log p$, and sample from the extrapolated policy $\pi^{\gamma}_{\text{extra}}$ only at those positions (substituting π_{RL} in Algo. 1, Line 6).

Empirical Setup. We evaluate our method on the AIME-24 benchmark using the ORZ, DAPO, and UniReason model pairs, generating 32 samples per question (see Appx. A.1 for more details). To isolate the impact of our strategy, we compare three approaches: (1) RLVR: The original, non-intervened RLVR model π_{RL}; (2) Selective Replace: Base model with tokens replaced by π_{RL}; (3) Selective Extrapolate: Base model with tokens replaced by $\pi^{\gamma}_{\text{Extra}}$. For a controlled comparison, we use the same selection criteria for (2) and (3), with the only difference being the extrapolation.

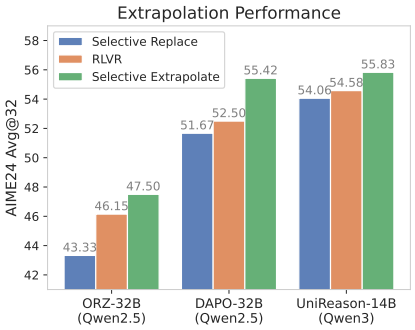


Figure 4: Extrapolation Performance

Results. On AIME-24, Selective Extrapolation yields higher Avg@32 (average of 32 samples) than π_{RL} across ORZ-32B, DAPO-32B, and UniReason-14B under matched gates (Fig. 4). In contrast, Selective Replace matches but does not surpass the RL baseline under the same criteria. These results indicate that moving beyond π_{RL} along $\Delta \log p$ provides incremental gains in reasoning accuracy.

Extrapolating on π_{RL}. We also apply selective extrapolation directly on π_{RL} rather than on π_{Base} in Algo. 1 (Line 4). As the threshold τ in f^{τ}_{logp} increases, the AIME-24 performance improves up to a moderate intervention ratio, after which gains plateau (Table 1). This pattern aligns with the sparsity finding: amplifying a limited set of reasoning-critical tokens is effective, whereas aggressive interventions yield diminishing returns.

Table 1: Selective Extrapolate ($\gamma = 0.1$) on the RLVR model (DAPO-32B) instead of the base model.

Replace Ratio	0.0%	1.8%	5.2%	20.0%
Avg@32	52.50	53.96	**55.31**	<u>55.10</u>
Threshold τ	N/A	-0.5	-0.2	0.0

Theoretical Justification. Following a standard simplification in theoretical analysis for LLM RL training (Munos et al., 2024; Shi et al., 2025), we consider a tabular softmax bandit policy: $\pi_\theta(y|x) \propto \exp(\theta_{x,y})$, where the logit is individually parameterized by θ for each prompt-response pair (x, y). We assume the policy is trained with Natural Policy Gradient (NPG (Kakade, 2001)) following Cui et al. (2025), since its updates resemble the controlled optimization of PPO (Schulman et al., 2017). The update rule of NPG via backtracking simplifies to: $\theta^{t+1}_{x,y} - \theta^{t}_{x,y} = \eta \cdot A^t(x, y)$, where η is the step size and A^t is the advantage function (Agarwal et al., 2021). In this context, our extrapolated policy (Eq. 7) is defined as $\pi_{\omega(\theta^t,\gamma)}$, where $\omega(\theta^t, \gamma) = \theta^t + \gamma(\theta^t - \theta^0)$. Under these conditions, we have the following theorem (the proof can be found in Appx. C):

Theorem 4.1. *For a given prompt x, if a tabular softmax policy π_{θ^t} is updated via natural policy gradient (Kakade, 2001), then the extrapolated policy $\pi_{\omega(\theta^t,\gamma)}$ satisfies:*

$$\exists\, \gamma > 0, \mathbb{E}_{y\sim\pi_{\omega(\theta^t,\gamma)}(\cdot|s)}[R_{x,y}] \geq \mathbb{E}_{y\sim\pi_{\theta^t}(\cdot|s)}[R_{x,y}].$$

Equality holds if and only if the reward $R_{x,y}$ is constant for all y.

This theorem shows that, in the simplified setting, extrapolating along the learned difference direction of $\Delta \log p$ can improve the expected reward. Nevertheless, we need to note that the proof relies on the idealized NPG's update rule, with a monotonic learning process consistently adjusting the logits along the reward's direction. In contrast, our empirical analysis has shown that the updates learned by RLVR concentrate only on a minority of tokens, with $\Delta \log p$ on most tokens being negligible. This disparity motivates our selective extrapolation only on positions with a significant difference, which exhibit the consistent, directional updates assumed by the theory.

4.2 Training-Time Enhancement via Advantage Reweighting

Training-time enhancement via probability-aware advantage reweighting. While our test-time approach amplifies the learned reasoning signal post-hoc, our training-time strategy proactively strengthens the model's reasoning signal during learning. Instead of extrapolating the final logp

Table 2: Comparison of our reweighting method and DAPO on math reasoning benchmarks.

Model	Method	AIME24		AIME25		AMC		Average	
		Avg@32	Pass@16	Avg@32	Pass@16	Avg@32	Pass@16	Avg@32	Pass@16
Qwen2.5-Math-7B	Base	14.79	47.46	6.67	27.84	40.62	79.25	20.69	51.52
	DAPO	35.73	54.09	17.6	30.45	73.04	89.03	42.12	57.86
	Ours	**39.06**	**60.58**	**18.54**	**36.72**	**73.64**	**89.69**	**43.75**	**62.33**
Qwen3-8B-Base	Base	5.42	30.63	5.73	32.8	27.64	78.09	12.93	47.17
	DAPO	36.98	**72.3**	26.67	46.76	69.13	88.51	44.26	69.19
	Ours	**38.13**	69.87	**31.15**	**55.38**	**71.05**	**92.3**	**46.78**	**72.52**

difference $\Delta \log p$, we leverage the observed correlation between high $\Delta \log p$ and low-probability tokens (Fig. 3(b)), and propose to amplify the learning signal of these critical low-probability tokens. Since the parameter update is driven by the advantage term $\hat{A}_{i,t}$ in policy gradient methods, we modify the advantage in DAPO (Eq. 4) to prioritize low-probability tokens:

$$\tilde{A}_{i,t} = \left[1 + \alpha \cdot \left(1 - \pi_{\theta_{\text{old}}}(y_{i,t}|x, y_{i,<t})\right)\right] \cdot \hat{A}_{i,t}, \tag{8}$$

where α is a hyperparameter controlling the reweighting strength. Such a concentration on low-probability tokens also aligns with our top-p experiment in Fig. 3(c), which finds that low-probability tokens are irreplaceable for RLVR training.

Experimental setup. We modify only the advantage (Eq. 8) in the standard DAPO recipe and keep all other hyperparameters fixed. We evaluate model performance on three math reasoning benchmarks: AIME-24, AIME-25, and AMC. Following DAPO's setup, we use top-p=0.7 for sampling during evaluation. We report Avg@32 and Pass@16[3], both computed over 32 samples per problem to ensure a stable estimate of the pass rates (Chen et al., 2021).

Results: performance gains across models and datasets. We compare our reweighting method on two models: Qwen2.5-Math-7B (Yang et al., 2024) and Qwen3-8B-Base (Yang et al., 2025a). As shown in Tab. 2, enhancing low-probability tokens' weight consistently improves reasoning accuracy across all tested models and datasets. Notably, this enhanced accuracy (Avg@32) doesn't come at the cost of exploration ability (often measured by Pass@k) (Yue et al., 2025); in fact, the average Pass@16 also increases over the DAPO baseline.

Comparison of different reweighting. While our reweighting method is motivated by the critical role of low-probability tokens, existing work has proposed alternative reweighting strategies that stem from different hypotheses: (1) PPL: Deng et al. (2025) find that RLVR updates favor *low-ppl* responses, so they reweight advantage to enhance these responses: $\tilde{A}^{\text{ppl}}_{i,t} = [1 - \alpha \cdot w_{\text{ppl}}(y_i)] \cdot \hat{A}_{i,t}$, where $w_{\text{ppl}}(y_i)$ is a normalized log-PPL weight. (2) Dominate: Yang et al. (2025b) argue that RLVR training can be *over-dominated* by low-probability tokens, so they propose to counteract this by upweighting high-probability tokens: $\tilde{A}^{\text{dom}}_{i,t} = [\alpha \cdot \pi_\theta(y_{i,t}) + 1 - \alpha] \cdot \hat{A}_{i,t}$. We implement these methods using their recommended hyperparameters and compare the performance on Qwen2.5-Math-7B. As shown in Table 3, our method of directly amplifying low-probability tokens achieves the best overall performance for both Avg@32 and Pass@16. The training dynamics in Fig. 5 provide further insight: Our method not only exhibits higher reasoning accuracy but also a steady increase in response length. This simultaneous increase in performance and length is a key pattern in effective reasoning RLVR training (Guo et al., 2025), suggesting the promoted reasoning behavior by our method. Moreover, the training entropy of $\tilde{A}^{\text{dom}}_{i,t}$ reweighting is clearly lower, since they adopt a

Table 3: Results of various reweighting methods.

Method		PPL	Dominate	Ours
AIME24	Avg@32	35.63	36.35	**39.06**
	Pass@16	**61.95**	55.27	60.58
AIME25	Avg@32	16.46	13.02	**18.54**
	Pass@16	32.19	20.69	**36.72**
AMC	Avg@32	72.06	**79.97**	73.64
	Pass@16	89.1	84.93	**89.69**
Average	Avg@32	41.38	43.11	**43.75**
	Pass@16	61.08	53.63	**62.33**

[3]With 32 samples, we report the more stable Pass@16 instead of Pass@32 for Pass@k evaluation.

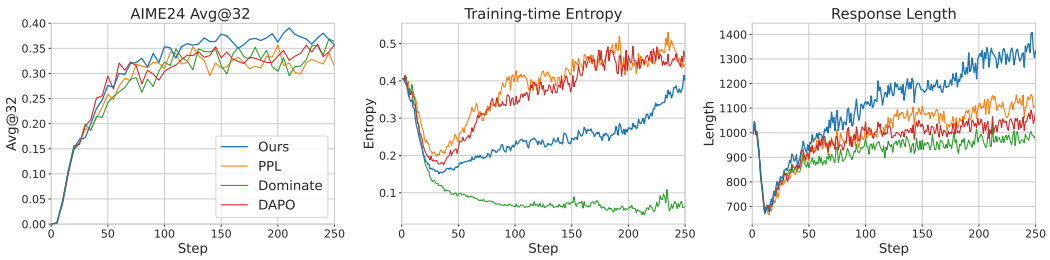


Figure 5: Training curves for different reweighting methods on Qwen2.5-Math-7B.

more restrictive clip-higher ratio of $\epsilon_{\text{high}} = 0.24$ than the default $\epsilon_{\text{high}} = 0.28$ in DAPO[4]. The lower entropy (less exploration) also explains their reduced Pass@k performance in Tab. 3.

5 Related Work

Reinforcement learning for LLM. Reinforcement learning is a pivotal component of the LLM post-training pipeline. Early applications centered on Reinforcement Learning from Human Feedback (RLHF) for model alignment (Ouyang et al., 2022; Stiennon et al., 2020), while recent advancements shift their focus to building reasoning models with RL. OpenAI o1 (Jaech et al., 2024) is the first reasoning model, and DeepSeek R1 (Guo et al., 2025) introduces a detailed RLVR (Lambert et al., 2024) recipe for building reasoning models with the GRPO algorithm (Shao et al., 2024). These seminal works inspired the development of a series of subsequent models, from industrial systems like Kimi(Team, 2025), Qwen3 (Yang et al., 2025a), and Gemini 2.5 (Comanici et al., 2025), to open-source academic algorithms such as Dr.GRPO (Liu et al., 2025), Open-Reasoner-Zero (Hu et al., 2025a), DAPO (Yu et al., 2025), and GSPO (Zheng et al., 2025), to further improve the reasoning abilities. In this paper, we adopt DAPO as our baseline RLVR algorithm.

Understanding the effects of RLVR. The success of RLVR has prompted a line of research dedicated to understanding its effects. While early work analyzed high-level cognitive behaviors of RLVR-trained models (Gandhi et al., 2025; Hu et al., 2025b; Bogdan et al., 2025), recent studies have deepened the analysis with token-level quantification (Qian et al., 2025; Wang et al., 2025a). For example, Cui et al. (2025) studied the token entropy change during RLVR, Yang et al. (2025b) quantified the gradient norm of specific tokens, and Deng et al. (2025) used token replacement to measure their impact on reasoning performance. A core finding from these analyses is that RLVR induces sparse updates, which have been verified through high-entropy tokens (Wang et al., 2025b), KL Divergences (Huan et al., 2025), and the sparse gradient norm (Yang et al., 2025b; Deng et al., 2025). However, these studies primarily focus on the magnitude of RLVR-induced updates while largely overlooking their direction, and our work centers on addressing this gap.

6 Conclusion

In this work, we introduced a directional analysis of RLVR based on the logp difference $\Delta \log p$, shown to be more effective in identifying sparse yet reasoning-critical updates than magnitude-based metrics (*e.g.,* divergence or entropy). Building on this, we proposed a test-time extrapolation to amplify these directional updates and a training-time reweighting to focus learning on the low-probability tokens that $\Delta \log p$ highlights. Both methods improve reasoning performance across different settings, validating our key principle: diagnose and improve RLVR by its update direction.

Limitations and future work. One primary limitation of our extrapolation method is the requirement of two models; future work could integrate this with parameter-efficient finetuning to reduce computational cost. The extrapolation also introduces additional hyperparameters, and future work can explore combining the selection threshold and extrapolation strength for a more adaptive extrapolation. Additionally, our reweighting approach could be evaluated for different model scales or combined with other adaptive training techniques.

[4]This follows the recommended value in their paper (Yang et al., 2025b). We also tested the default $\epsilon_{\text{high}} = 0.28$, but it resulted in unstable training.

Reproducibility Statement

To ensure the reproducibility of our work, we provide detailed descriptions of our experimental setup, including necessary implementation details and hyperparameter settings in the appendix. We'll also release our source code publicly upon acceptance.

References

Alekh Agarwal, Sham M. Kakade, Jason D. Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research*, 22(98):1–76, 2021. URL http://jmlr.org/papers/v22/19-736.html.

Paul C. Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. Thought anchors: Which llm reasoning steps matter?, 2025. URL https://arxiv.org/abs/2506.19143.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, et al. Evaluating large language models trained on code, 2021.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Jia Deng, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, and Ji-Rong Wen. Decomposing the entropy-performance exchange: The missing keys to unlocking effective reinforcement learning, 2025.

Kanishk Gandhi, Ayush K Chakravarthy, Anikait Singh, Nathan Lile, and Noah Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective STars. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=QGJ9ttXLTy.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638, 2025.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025a.

Xiao Hu, Xingyu Lu, Liyuan Mao, YiFan Zhang, Tianke Zhang, Bin Wen, Fan Yang, Tingting Gao, and Guorui Zhou. Why distillation can outperform zero-rl: The role of flexible reasoning. *arXiv preprint arXiv:2505.21067*, 2025b.

Maggie Huan, Yuetai Li, Tuney Zheng, Xiaoyu Xu, Seungone Kim, Minxin Du, Radha Poovendran, Graham Neubig, and Xiang Yue. Does math reasoning improve general llm capabilities? understanding transferability of llm reasoning. *arXiv preprint arXiv:2507.00432*, 2025.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Sham M Kakade. A natural policy gradient. In T. Dietterich, S. Becker, and Z. Ghahramani (eds.), *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001.

Maxim Khanov, Jirayu Burapacheep, and Yixuan Li. ARGS: Alignment as reward-guided search. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=shgx0eqdw6.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares-López, Jessica Hoffmann, Lucas Dixon, Michal Valko, and Mathieu Blondel. Decoding-time realignment of language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 31015–31031. PMLR, 21–27 Jul 2024.

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. *arXiv preprint arXiv:2503.20783*, 2025.

Remi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Côme Fiegel, et al. Nash learning from human feedback. In *Forty-first International Conference on Machine Learning*, 2024.

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

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in llm reasoning. *arXiv preprint arXiv:2506.02867*, 2025.

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *The Thirteenth International Conference on Learning Representations*, 2025.

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

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

Ruizhe Shi, Runlong Zhou, and Simon Shaolei Du. The crucial role of samplers in online direct preference optimization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=F6z3utfcYw.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:3008–3021, 2020.

Kimi Team. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Haozhe Wang, Qixin Xu, Che Liu, Junhong Wu, Fangzhen Lin, and Wenhu Chen. Emergent hierarchical reasoning in llms through reinforcement learning. *arXiv preprint arXiv:2509.03646*, 2025a.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, et al. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*, 2025b.

Yuancheng Xu, Udari Madhushani Sehwag, Alec Koppel, Sicheng Zhu, Bang An, Furong Huang, and Sumitra Ganesh. GenARM: Reward guided generation with autoregressive reward model for test-time alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=J0qTpmbSbh.

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, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Zhihe Yang, Xufang Luo, Zilong Wang, Dongqi Han, Zhiyuan He, Dongsheng Li, and Yunjian Xu. Do not let low-probability tokens over-dominate in rl for llms. *arXiv preprint arXiv:2505.12929*, 2025b.

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

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

A Selective Token Replacement & Extraploation

A.1 Implementation Details

Models. Our experiments use several publicly available RLVR-trained models and their corresponding base models from the Qwen series (Yang et al., 2025a; Team, 2024):

- ORZ: The Open-Reasoner-Zero-32B model (Hu et al., 2025a), finetuned from Qwen2.5-32B base model using the PPO algorithm.
- DAPO: The DAPO-Qwen-32B model (Yu et al., 2025), finetuned from the same Qwen2.5-32B base but with the DAPO algorithm.
- UniReason: The UniReason-Qwen3-14B-RL model (Huan et al., 2025), finetuned from Qwen3-14B-Base using the GRPO algorithm.

Sampling settings. We utilize the AIME-24 dataset to evaluate the replacement performance. We adopt the default chat prompt template from each model, with the user prompt defined as follows:

```
[Question]
Please reason step by step, and put your final answer within \\boxed{}.
```

We set the sampling parameters with top-p=0.7, temperature=1.0, max-length=20k, and sample 32 responses for each question. The answer is extracted from the last "boxed" wrapped text and verified using Math-Verify. We report the correctness averaged over 32 samples, *i.e.,* Avg@32.

Hyperparameters for extrapolation. As described in Algo. 1, the replacement is adopted selectively, controlled by the threshold τ in the criteria function f^{τ}, while the extrapolation strength is adjusted by the parameter γ in $\pi^{\gamma}_{\text{Extra}}$. For the extrapolation results in Fig. 4, the "Selective Extrpolate" and "Selective Replace" methods share the same hyperparameters for each model, which we summarize as follows:

Table 4: Hyperparameters for the extrapolation results (Fig. 4).

Model	ORZ	DAPO	UniReason
Threshold τ for f^{γ}_{logp}	-0.4	-0.35	-0.3
Replaced Ratio	10.1%	7.5%	11.4%
γ in $\pi^{\gamma}_{\text{Extra}}$	0.1	0.1	0.05

A.2 Additional Experiments

Additional metrics. As described in Sec. 3, our primary metrics for token replacement are the base model's entropy $\mathcal{H}_{\text{Base}}$, KL Divergence $\mathbb{D}^{\text{KL}}$, and logp difference $\Delta \log p$. For our ablation study, we include additional metrics: the RLVR model's entropy $\mathcal{H}_{\text{RL}}$ and two KL-divergence variants: $\mathbb{D}^{\text{KL}}_{\pi_{\text{RL}},\pi_{\text{Base}}}$ and $\mathbb{D}^{\text{KL}}_{\pi_{\text{Base}},\pi_{\text{RL}}}$. We evaluate these metrics as criteria for the DAPO model's selective replacement. By varying the threshold τ for each criterion, we control the token replacement frequency and plot the performance on AIME-24 against various replacement ratios in Fig. 6. As shown in the figure, although the additional metrics' selected replacements also approach the RLVR model's performance, they still require more replacement than $\Delta \log p$ does. This confirms the performance ordering for identifying reasoning-critical tokens: logp difference $>$ divergence $>$ entropy.

Selected Tokens. To provide an intuitive comparison of the metrics, we analyze the tokens utilized for replacing the base model's choice during DAPO's token replacement of entropy $\mathcal{H}_{\pi_{\text{Base}}}$, KL Divergence $\mathbb{D}^{\text{KL}}$, and logp difference $\Delta \log p$. To ensure a fair comparison, we adjust the threshold for each metric to achieve a replacement rate of approximately 8%. Figure 7 illustrates each criterion's top 50 substitution tokens. The figure reveals that entropy-based selection favors logical transition words (e.g., Thus, need, can), while the divergence and $\Delta \log p$ criteria utilize more specific mathematical reasoning tokens, including a higher proportion of math symbols. Combined with the inferior performance of the entropy criterion, this suggests that these specific mathematical tokens might be more efficient for improving reasoning performance.

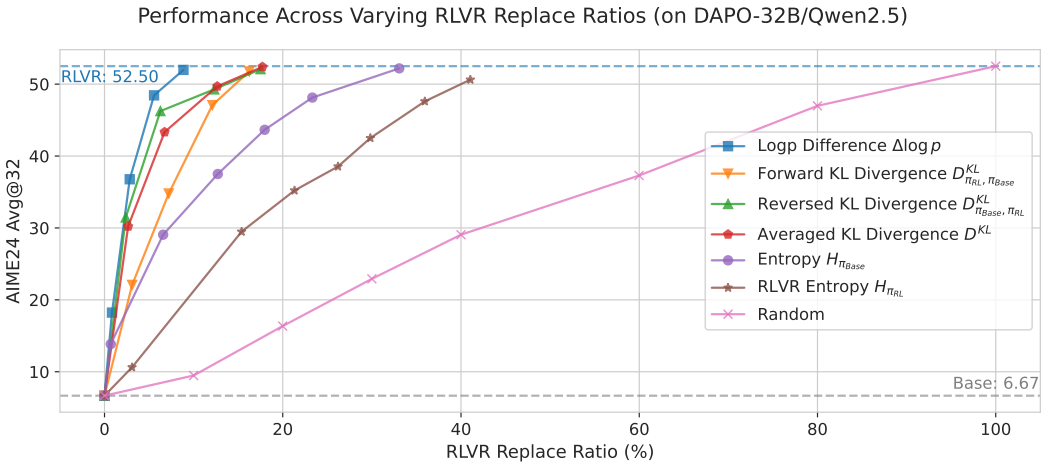


Figure 6: Selective token replacement results with additional criteria for DAPO.

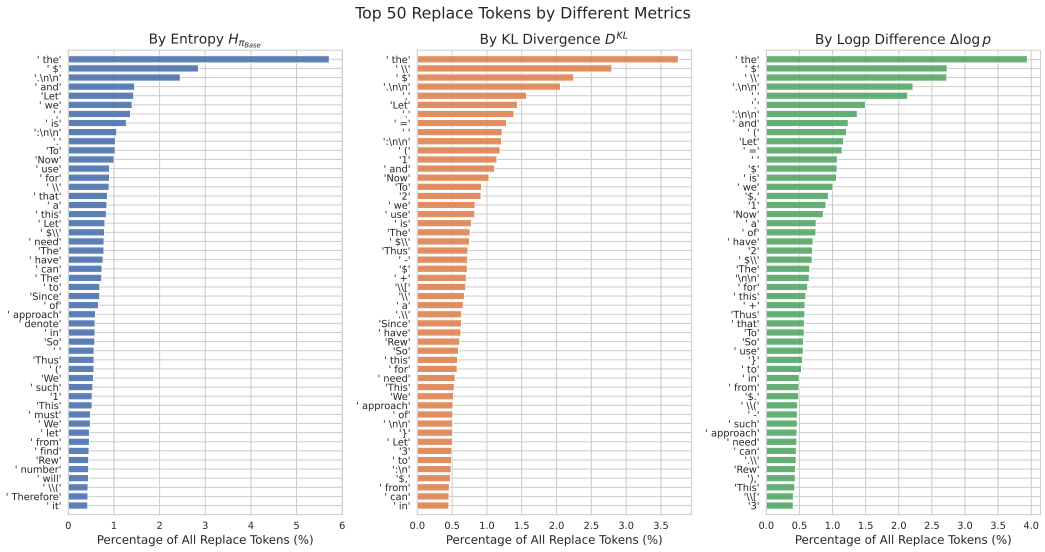


Figure 7: Top 50 tokens for replacing the base model's choice under different metrics' selection.

B RLVR Training Setting

We adopt the open-sourced DAPO recipe for RLVR training. Our configuration includes double clip ratios ($\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$) and a learning rate of 1e-6 with a 10-step warmup. Each RLVR step consists of 512 prompts with 16 sampled responses each, processed in mini-batches of 32 prompts to yield 16 gradient updates per step. Maximum generation length and overlong penalty thresholds are set to 8k/4k for Qwen2.5-Math-7B and 20k/16k for Qwen3-8b-base, respectively.

For reweighting, our parameter α (Eq. 8) is set to 0.2 for Qwen2.5 and 0.1 for Qwen3. Following the recommended values by Deng et al. (2025) and Yang et al. (2025b), we set α to 0.1 for $\tilde{A}^{\text{dom}}_{i,t}$ and 0.01 for $\tilde{A}^{\text{PPL}}_{i,t}$. For $\tilde{A}^{\text{dom}}_{i,t}$ specifically, we also adjust ϵ_{high} to 0.24.

C Proofs

Proof of Lemma 3.1. For ease of notation, we omit the context $x, y_{i,<t}$ here. The derivative of DAPO on an unclipped token $y_{i,t}$ is:

$$\begin{aligned}\nabla_\theta \mathcal{J}_{\text{DAPO}}(y_{i,y}) = \nabla_\theta\, r_{i,t}(\theta)\hat{A}_{i,t} &= \nabla_\theta\, \frac{\pi_\theta(y_{i,t})}{\pi_{\theta_{\text{old}}}(y_{i,t})}\hat{A}_{i,t} \\ &= r_{i,t}(\theta)\hat{A}_{i,t} \cdot \nabla_\theta \log \pi_\theta(y_{i,t}) \\ &= w_{i,t} \cdot \nabla_\theta \log \pi_\theta(y_{i,t}).\end{aligned}$$

For the softmax-parameterized policy π_θ with logits z for $y_{i,t}$, assuming $y_{i,t}$ corresponds to index k of vocabulary $\mathcal{V}$, we have:

$$\begin{aligned}\frac{\partial}{\partial z_j}\log \pi_\theta(y_{i,t}) &= \frac{1}{\pi_\theta(y_{i,t})} \cdot \frac{\partial}{\partial z_j}\frac{\exp(z_k)}{\sum_l \exp(z_l)} \\ &= \frac{1}{\pi_\theta(y_{i,t})} \cdot \begin{cases} \frac{\exp(z_k)\sum_l \exp(z_l) - \exp(z_k)\exp(z_k)}{(\sum_l \exp(z_l))^2}, & j = k \\ \frac{-\exp(z_k)\exp(z_j)}{(\sum_l \exp(z_l))^2}, & j \neq k \end{cases} \\ &= \begin{cases} 1 - \pi_\theta(\mathcal{V}_k), & j = k \\ -\pi_\theta(\mathcal{V}_j), & j \neq k \end{cases} \\ &= \mathbb{I}(j = k) - \pi_\theta(\mathcal{V}_j).\end{aligned}$$

So the $\ell 1$-norm of $\nabla_z \mathcal{J}_{\text{DAPO}}(y_{i,t})$ becomes:

$$\begin{aligned}\|\nabla_z \mathcal{J}_{\text{DAPO}}(y_{i,t})\|_1 &= \|w_{i,t}\nabla_z \log \pi_\theta(y_{i,t})\|_1 \\ &= |w_{i,t}| \cdot \sum_j \Big|\mathbb{I}(j = k) - \pi_\theta(\mathcal{V}_j)\Big| \\ &= |w_{i,t}| \cdot \Big(1 - \pi_\theta(y_{i,t}) + \sum_{j \neq k} \pi_\theta(\mathcal{V}_j)\Big) \quad (y_{i,t} = \mathcal{V}_k) \\ &= |w_{i,t}| \cdot 2\big(1 - \pi_\theta(y_{i,t})\big).\end{aligned}$$

□

Proof of Theorem 4.1. Let $\mathcal{J}(\theta_x) = \mathbb{E}_{y \sim \pi_{\theta_x}(\cdot)}[R_{x,y}]$, and we need to show that for each x:

$$\exists\, \gamma > 0, \mathcal{J}(\theta^t_x + \gamma(\theta^t_x - \theta^0_x)) \geq \mathcal{J}(\theta^t_x).$$

Denote the extrapolation direction as $d^t_x = \theta^t_x - \theta^0_x$, this is equivalent to showing the directional derivative of $\mathcal{J}$ at θ^t_x along d^t_x is positive.

The directional derivative is given by:

$$\nabla_{d^t_x}\mathcal{J}(\theta^t) = \nabla_{\theta_x}\mathcal{J}(\theta^t_x)^\top \frac{d^t_x}{\|d^t_x\|} = \frac{1}{\|d^t_x\|} \cdot \sum_y \frac{\partial \mathcal{J}(\theta^t_x)}{\partial \theta_{x,y}} d^t_{x,y}.$$

For the softmax policy $\pi_{\theta_x}(y) = \exp(\theta_{x,y})/\sum_{y'}\exp(\theta_{x,y'})$, its gradient satisfies:

$$\frac{\partial \pi_{\theta_x}(y')}{\partial \theta_{x,y}} = \pi_{\theta_x}(y')\left(\mathbb{I}(y=y') - \pi_{\theta_x}(y)\right).$$

So the partial gradient of $\mathcal{J}$ on y is:

$$\frac{\partial \mathcal{J}(\theta_x)}{\partial \theta_{x,y}} = \sum_{y'} R_{x,y'}\frac{\partial \pi_{\theta_x}(y')}{\partial \theta_{x,y}} = R_{x,y}\pi_{\theta_x}(y) - \pi_{\theta_x}(y)\sum_{y'} R_{x,y'}\pi_{\theta_x}(y') = \pi_{\theta_x}(y)(R_{x,y} - \pi_{\theta_x}^\top R_x).$$

Note that the advantage is $A^t(x,y) = R_{x,y} - \pi_{\theta_x^t}^\top R_x$ under the bandit setting, the directional derivative thus becomes:

$$\begin{aligned}\nabla_{d_x^t}\mathcal{J}(\theta^t) &= \frac{1}{\|d_x^t\|}\cdot\sum_y \pi_{\theta_x^t}(y)(R_{x,y} - \pi_{\theta_x^t}^\top R_x)d_{x,y}^t \\ &= \frac{1}{\|d_x^t\|}\cdot\sum_a \pi_{\theta_x^t}(y)\cdot A^t(x,y)\cdot d_{x,y}^t\end{aligned}$$

We now analyze the order of $A^t(x,y)$ and $d_{x,y}^t$.

Under the assumed bandit setting, the order of $A^t(x,y)$ is the same as the order of $R_{x,y}$, i.e., $A^t(x,y_1) > A^t(x,y_2)$ if and only if $R_{x,y_1} > R_{x,y_2}$. For $d_{x,y}^t$, we can prove that its order is also the same as $R_{x,y}$ with induction.

At $t=1$, using the update rule of NPG, we have:

$$d_{x,y}^1 - d_{x,y'}^1 = \eta\cdot(A^0(x,y) - A^0(x,y')) = \eta\cdot(R_{x,y} - R_{x,y'}).$$

So the order of $d_{x,y}^1$ is the same as $R_{x,y}$. Assume at iteration t, the order of $d_{x,y}^t$ is the same as $R_{x,y}$, then at iteration $t+1$, we have:

$$d_{x,y}^{t+1} - d_{x,y'}^{t+1} = d_{x,y}^t - d_{x,y'}^t + \eta\cdot(A^t(x,y) - A^t(x,y')) = d_{x,y}^t - d_{x,y'}^t + \eta\cdot(R_{x,y} - R_{x,y'}).$$

So we still have $d_{x,y}^{t+1} > d_{x,y'}^{t+1} \iff R_{x,y} > R_{x,y'}$. Thus by induction, the order of $d_{x,y}^t$ is the same as $R_{x,y}$ for all t.

Since the order of $A^t(x,y)$ and $d_{x,y}^t$ are the same, we can apply the Chebyshev sum inequality to get:

$$\sum_y \pi_{\theta_x^t}(y)\cdot\sum_y \pi_{\theta_x^t}(y)\cdot A^t(x,y)\cdot d_{x,y}^t \geq \left(\sum_y \pi_{\theta_x^t}(y)\cdot A^t(x,y)\right)\cdot\left(\sum_y \pi_{\theta_x^t}(y)\cdot d_{x,y}^t\right),$$

with the equality holds if and only if $A^t(x,y)$ or $d_{x,y}^t$ is a constant for all y (i.e., constant reward).

Note that the expectation of advantage $\sum_y \pi_{\theta_x^t}(y)\cdot A^t(x,y) = 0$, so we have:

$$\nabla_{d_x^t}\mathcal{J}(\theta^t) = \frac{1}{\|d_x^t\|}\cdot\sum_y \pi_{\theta_x^t}(y)\cdot A^t(x,y)\cdot d_{x,y}^t \geq 0.$$

The equality holds if and only if $R_{x,y}$ is a constant for all y.

□

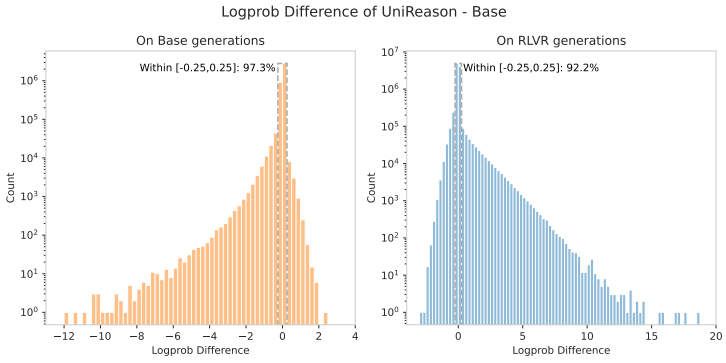
D Statistical Comparison of Different Metrics

Empirical setup. We evaluate three RLVR models: ORZ, DAPO, UniReason, and their base counterparts. For each model, we generate 32 responses per question from the AIME-24 dataset, with a sampling strategy of top-p=0.7 and temperature=1.0. Our analysis focuses on several metrics comparing the model pairs: the base/RLVR model's entropy, KL divergences, and the logp difference. The probability distribution versus different $\Delta\log p$ bins in Fig. 3(b) is also measured on the DAPO's generation under this setting.

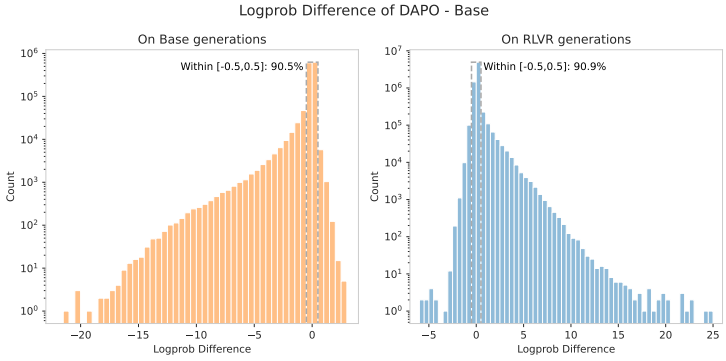
Statistics of Different Metrics. We compute each metric of the three RLVR model pairs on both the base model and the RLVR model's generation. As shown in Fig. 8, the distribution of logp difference $\Delta\log p$ is bimodal, with a positive tail for the RLVR's generated text and a negative tail for the base model's generation. In contrast, the distributions of other magnitude-based metrics are nearly identical regardless of which model generated the output (Fig. 9-11).

E THE USE OF LARGE LANGUAGE MODELS

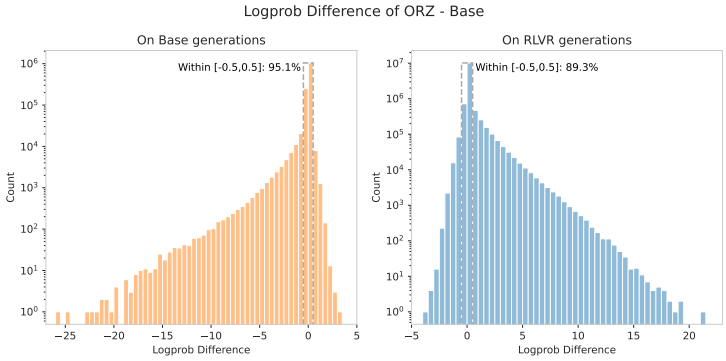
We utilize LLMs only to polish some of the language of this paper. All content was originally drafted by the authors. The use of LLMs was restricted to refining some pre-existing text, and any suggested modifications were reviewed by the authors to confirm their accuracy and alignment with the original meaning.



(a) Logp difference of UniReason

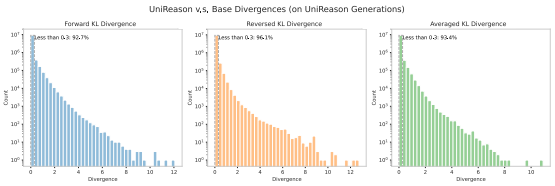


(b) Logp difference of DAPO

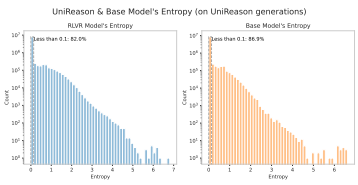


(c) Logp difference of ORZ

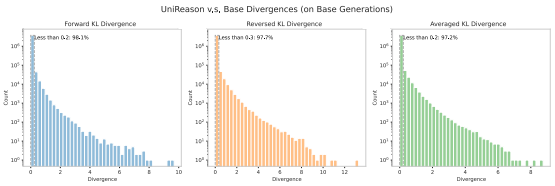
Figure 8: Logp Difference histograms of different RLVR models, comparing the RLVR and base model's generations.



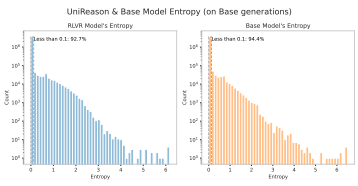
(a) Divergence on UniReason's generations



(b) Entropy on UniReason's generations

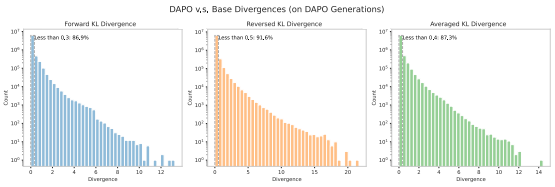


(c) Divergence on base's generations

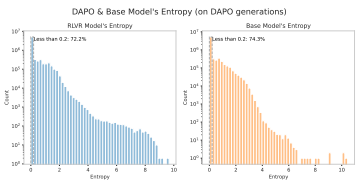


(d) Entropy on base's generations

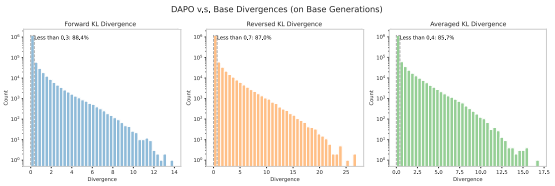
Figure 9: Divergence and entropy histograms of UniReason and its corresponding base model measured on UniReason or the base model's generations.



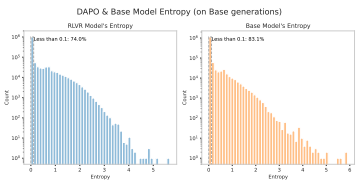
(a) Divergence on DAPO's generations



(b) Entropy on DAPO's generations

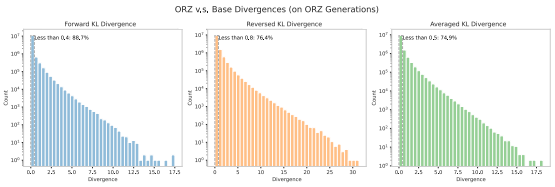


(c) Divergence on base's generations

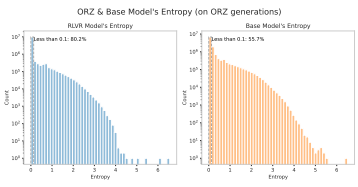


(d) Entropy on base's generations

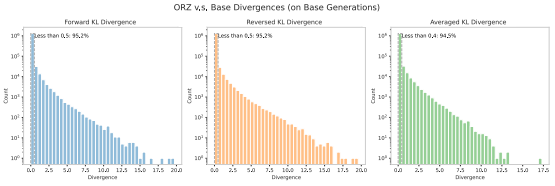
Figure 10: Divergence and entropy histograms of DAPO and its corresponding base model measured on DAPO or the base model's generations.



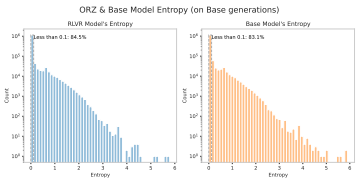
(a) Divergence on ORZ's generations



(b) Entropy on ORZ's generations



(c) Divergence on base's generations



(d) Entropy on base's generations

Figure 11: Divergence and entropy histograms of ORZ and its corresponding base model measured on ORZ or the base model's generations.