

Rebellious Student: Reversing Teacher Signals for Reasoning Exploration with Self-Distilled RLVR

Jeonghye Kim^{1,2*◊}, Jiwon Jeon^{2*}, Dongsheng Li¹, Yuqing Yang^{1†}

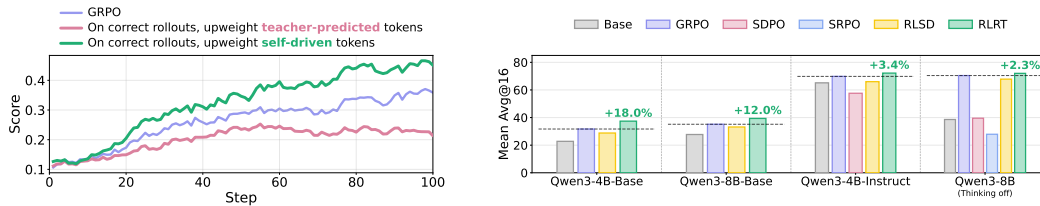
¹Microsoft Research ²KAIST

{jeonghye.kim, jiwon.jeon}@kaist.ac.kr, {dongshli, yuqyang}@microsoft.com

Abstract

Self-distillation has emerged as a powerful framework for post-training LLMs, where a teacher conditioned on extra information guides a student without it, both from the same model. While this guidance is useful when the student has failed, on successful rollouts, the same mechanism instead overwrites the student’s choices and suppresses its own reasoning. Therefore, we propose reading the original self-distillation signal in reverse: when the student succeeds along a path the teacher would not have predicted, these tokens reflect its *self-driven reasoning*. Building on this, we propose RLRT (RLVR with Reversed Teacher), which augments GRPO by reinforcing these tokens on correct rollouts. We interpret this as a new form of exploration in RLVR: not uniform diversity, but valuable exploration grounded in the student’s own success. Across base, instruction-tuned, and thinking-tuned Qwen3 checkpoints, RLRT substantially outperforms self-distillation and exploration-based baselines, establishing information asymmetry as a new, principled design axis for RLVR.

1 Introduction



(a) Training reward on Qwen3-4B-Base by token weighting

(b) Mean benchmark score across models and methods

Figure 1: Reversing the teacher signal turns self-distillation into valuable exploration. (a) Training reward on Qwen3-4B-Base under GRPO, upweighting **teacher-predicted** tokens on correct rollouts, and upweighting **self-driven** tokens (RLRT). (b) Mean avg@16 score over six math benchmarks (AIME24/25/26, HMMT26, AMC23, MATH500) across four Qwen3 backbones. RLRT consistently outperforms baselines significantly. Full results are in Tables 1 and 3. We skipped detailed comparisons with SDPO/SRPO on the base models because they collapsed early during training (Appendix F.2).

Reinforcement learning with verifiable rewards (RLVR) has become the dominant paradigm for post-training LLMs on reasoning tasks [6, 19], yet it suffers from a credit-assignment bottleneck: the only learning signal is a sparse scalar reward at the end of each trajectory. Self-distillation has recently emerged as a powerful response [9, 20, 32, 25]. Its core mechanism is an *information asymmetry* between two views of the same model: a *teacher* view conditioned on additional information (rich

*Equal contribution. ◊Work done during an internship at Microsoft Research. † Corresponding author.

textual feedback, or a successful peer rollout) and a *student* view without it. By distilling the teacher into the student, this asymmetry converts the sparse scalar reward into dense token-level supervision.

However, the value of distilling the teacher into the student depends on whether the rollout was already correct. On failed trajectories, conditioning the teacher on corrective information is useful: the teacher points the student toward solutions it could not previously reach on its own, and distillation transfers that corrective signal token by token. On already-successful trajectories, the same mechanism inverts its role. Even when the student already reached the correct answer, distilling toward the teacher overwrites the student’s choices with the teacher’s, a problem recently identified as *optimization ambiguity* in self-distillation [12]. Rather than being corrected, the student is forced to imitate a path it had already solved its own way, undermining the independent reasoning that produced the success.

This observation motivates us to **reverse the direction** of self-distillation on correct rollouts. Consider the tokens where the student’s choice differs most sharply from what the teacher would have predicted. On a correct rollout, these are not arbitrary disagreements. They are the very points where the student exercised its own reasoning, choosing against the teacher and still arriving at the correct answer. Such tokens carry the student’s *self-driven reasoning*: choices that succeeded despite going against the teacher. Therefore, rather than suppressing them by aligning the student to the teacher, we propose to amplify these self-driven tokens during training. In this way, self-distillation becomes a tool for strengthening the student’s reasoning ability, rather than reducing it to imitation.

This perspective also suggests a new angle for tackling the loss of reasoning diversity, a persistent failure mode of RLVR in which probability mass concentrates on trajectories the policy already prefers [29]. Existing methods address this through token-level entropy regulation [4, 18, 7] or sequence-level diversity objectives [8, 23, 21], broadening exploration in the hope that wider sampling will surface correct paths. However, they treat diversity as a uniform target, leaving the RL signal to decide which alternative choices are worth keeping. We take a different stance. Rather than encouraging diversity for its own sake, we identify, within the rollouts the model has already produced, tokens that are simultaneously self-driven (departing from the conditioned teacher) and verified (occurring on correct trajectories), and upweight them during training. This yields what we term *valuable exploration*: diversity grounded in successful reasoning rather than surface variation.

Building on this, we propose RLRT (RLVR with Reversed Teacher), which augments GRPO by reversing the direction of self-distillation on correct rollouts: instead of pulling the student to imitate the teacher, RLRT amplifies the self-driven tokens where the student reasoned differently from the teacher and still reached the correct answer. As shown in Figure 1, across Qwen3-4B/8B-Base, Qwen3-4B-Instruct, and Qwen3-8B, RLRT exhibits faster training-score growth and outperforms self-distillation baselines by an average of **8.9%** on six math reasoning benchmarks, including the challenging AIME and HMMT. We summarize our contributions as follows:

- **A new analysis.** We reinterpret the teacher–student gap on correct rollouts: prior self-distillation reads it as an alignment target pulling the student to imitate the teacher, whereas we show that, read in reverse, it localizes the student’s own *self-driven reasoning*.
- **A new algorithm.** Guided by this analysis, we propose **RLRT**, which augments GRPO by amplifying these self-driven tokens on correct rollouts, yielding consistent gains over strong RLVR baselines across base, instruction-tuned, and thinking-tuned models.
- **A broader implication.** Beyond a specific algorithm, our findings establish *information asymmetry* as a principled, intrinsic source of *valuable exploration*, offering a new design axis for RLVR.

2 Related Works

2.1 Self-Distillation in LLM Post-Training

A growing line of work improves LLM reasoning through information asymmetry within a single model acting as both teacher and student, where the teacher is conditioned on privileged context. This context takes diverse forms: ground-truth reasoning traces [32], runtime errors or judge evaluations as textual feedback [9, 13], second-turn revisions conditioned on critiques [22], expert demonstrations [20], and prepended in-context knowledge or system prompts [27]. Across these variants, the design intent is alignment: the teacher–student gap is used to pull the student toward the teacher, whether by matching distributions [32, 20, 27], distilling improved second-turn behavior into single-

turn [22], weighting tokens by the magnitude of teacher influence under verifiable rewards [25], or restricting alignment to failed rollouts only [12].

RLRT shares this asymmetric setup but inverts the alignment intent altogether: rather than pulling the student toward the teacher, we use the teacher–student gap in the opposite direction, treating tokens where the student diverged from the teacher on correct rollouts as evidence of self-driven reasoning, that is, choices made against the teacher’s prediction that nonetheless reached the correct answer.

2.2 Reasoning Exploration and Diversity

RLVR is widely observed to suffer from reasoning boundary collapse, where the policy concentrates on a narrow set of high-reward strategies rather than expanding its reasoning capacity [29, 17, 26]. Existing remedies broaden output diversity at two scales: token-level entropy regulation [4, 18, 24, 7, 3, 10] and sequence- or outcome-level objectives over full reasoning traces [8, 23, 21, 2, 5]. Both treat diversity as a uniform target and rely on local stochasticity or heuristic proxies such as embedding similarity, n-gram overlap, or outcome counts, capturing surface variation rather than meaningful reasoning differences.

RLRT takes a different route. Rather than treating diversity as a uniform target, it identifies, within already-correct rollouts, the specific tokens at which the student departed from the teacher and yet still reached the correct answer, yielding valuable exploration: diversity grounded in the student’s own successful reasoning rather than heuristic surrogates of variation.

3 Preliminaries

Notation. Let x be a prompt and $y = (y_1, \dots, y_T)$ a response from policy π_θ , with prefix $y_{<t} := (y_1, \dots, y_{t-1})$ and suffix $y_{>t} := (y_{t+1}, \dots, y_T)$. We write $h_t := (x, y_{<t})$ for the prefix history, $R \in \{0, 1\}$ for the verifiable reward, and \mathcal{V} for the vocabulary.

Self-distillation in RLVR. In RLVR with self-distillation, a single model serves as both student and teacher: the student conditions only on h_t , while the teacher additionally conditions on a privileged context c (e.g., the ground-truth solution or a successful rollout) hidden from the student [32, 9, 25]. We write

$$P_S^t(\cdot) := \pi_\theta(\cdot | h_t), \quad P_T^t(\cdot) := \pi_\theta(\cdot | h_t, c), \quad (1)$$

yielding a token-level log-probability ratio $\Delta_t := \text{sg}(\log P_T^t(y_t) - \log P_S^t(y_t))$, which measures how much the privileged context c revises the model’s belief about token y_t , with $\text{sg}(\cdot)$ denoting stop-gradient.

Distribution-matching approaches such as on-policy self-distillation (OPSD) [32] use Δ_t to drive P_S^t toward P_T^t directly. RLSD [25] observes that distribution matching is ill-posed when the student lacks access to c , since the target conditions on c while the student does not. To avoid this, RLSD repurposes the ratio as a magnitude-only credit signal, yielding the RLSD update

$$w_t^{\text{RLSD}} = \exp(\text{sign}(A) \cdot \Delta_t) = \left(\frac{P_T^t(y_t)}{P_S^t(y_t)} \right)^{\text{sign}(A)}, \quad (2)$$

where A is the group-relative advantage. The $\text{sign}(A)$ exponent ensures direction-aware credit assignment: on correct rollouts, tokens with $P_T^t > P_S^t$ are amplified (the teacher *favours* them); on incorrect rollouts, the same tokens are attenuated. Thus, the verifiable reward determines the sign of the update, while the teacher only modulates magnitude across tokens within a trajectory.

4 Motivation

In RLVR, meaningful reasoning gains come not from rollouts that merely reach the correct answer, but from those that arrive there through novel paths, ones that diverge from the model’s prior reasoning patterns. The teacher–student setup above provides a natural lens for identifying such moments. On correct rollouts, the tokens at which the student departs from the teacher are not merely mistakes to be suppressed, but signs of *self-driven reasoning*. More formally, we identify self-driven reasoning with tokens at which the student deviates from the teacher’s predictive distribution in ways influential

to reaching the correct answer. Such tokens are what push the student toward stronger reasoning, and in this section we discuss how to detect and reinforce them.

4.1 Information Asymmetry as an Exploration Signal

To analyse self-driven reasoning, we define the *token-level information asymmetry* \hat{D}_t at a sampled token y_t and the *position-level information asymmetry* \bar{D}_t as its expectation under the student:

$$\hat{D}_t(y_t) := \log \frac{P_S^t(y_t)}{P_T^t(y_t)}, \quad \bar{D}_t := \mathbb{E}_{v \sim P_S^t}[\hat{D}_t(v)] = \text{KL}(P_S^t \parallel P_T^t). \quad (3)$$

We claim that \bar{D}_t flags *which positions matter*, while the sign of \hat{D}_t marks *in which direction* the policy should update.

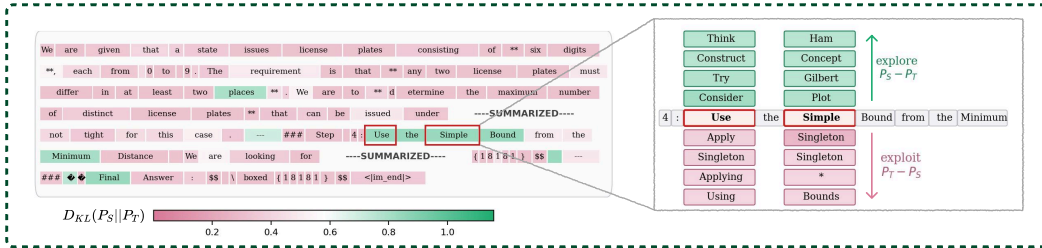


Figure 2: **Critical positions and explore/exploit directions.** Token shading shows the position-level information asymmetry $\bar{D}_t = \text{KL}(P_S^t \parallel P_T^t)$. At each critical position (right panels), candidate tokens are taken as the union of the teacher’s and student’s top-100 tokens; we display the top four with the largest $P_S^t - P_T^t$ (green, $\hat{D}_t > 0$) and the top four with the largest $P_T^t - P_S^t$ (pink, $\hat{D}_t < 0$).

Figure 2 illustrates \bar{D}_t and \hat{D}_t on a reasoning trajectory. Most tokens have small \bar{D}_t , but a few high-asymmetry tokens mark *critical positions* where token choice strongly affects the outcome. At these positions, candidates the teacher would have predicted ($\hat{D}_t < 0$, e.g., *use, conclude*) define the *exploit* direction, while candidates the student chose against the teacher’s prediction ($\hat{D}_t > 0$, e.g., *try, consider*) define the *explore* direction. Additional rollouts exhibiting the same pattern are provided in Appendix E. In the following subsections, we examine \bar{D}_t and \hat{D}_t in more detail.

4.2 \bar{D}_t Identifies Which Positions Matter

Claim. The position-level information asymmetry \bar{D}_t is large precisely at positions where the choice of token meaningfully affects the probability of a correct outcome.

Theoretical Justification. We justify the claim through a Bayesian view of the teacher. We model the teacher as π_θ conditioned on the event $R = 1$ (success), so that the student and teacher distributions become

$$P_S^t(\cdot) := \pi_\theta(\cdot \mid h_t), \quad P_T^t(\cdot) := \pi_\theta(\cdot \mid h_t, R = 1). \quad (4)$$

For each token $v \in \mathcal{V}$, let

$$f(v) := \Pr_{Y \sim P_S^t}[R = 1 \mid h_t, y_t = v], \quad \bar{f}_S^t := \mathbb{E}_{v \sim P_S^t}[f(v)],$$

denote the per-token correctness probability and its student-mean. Bayes’ rule then yields a single identity that underlies the analysis below.

Lemma 1 (Bayesian teacher). *At each step t , $P_T^t(v) = \frac{P_S^t(v)f(v)}{\bar{f}_S^t} \iff \hat{D}_t(v) = \log \bar{f}_S^t - \log f(v)$.*

The proof is deferred to Appendix C.1. The teacher is the student tilted toward tokens with higher $f(v)$; equivalently, $\hat{D}_t(v)$ measures how far $f(v)$ falls below \bar{f}_S^t .

In RLVR, any policy update at position t acts only on tokens the student actually samples, so the relevant signal is how much f varies among such tokens. We call this the *influence* of position t :

$$\text{Inf}_S(t) := \mathbb{E}_{v \sim P_S^t} [|f(v) - \bar{f}_S^t|]. \quad (5)$$

A position is *critical* when $\text{Inf}_S(t)$ is large and *inert* when near zero. While $\hat{D}_t(y_t)$ acts pointwise, its student-expectation $\bar{D}_t = \text{KL}(P_S^t \| P_T^t)$ from (3) captures the per-position effect of reweighting. The two scales are tied by a Pinsker-type bound.

Theorem 2 (\bar{D}_t controls $\text{Inf}_S(t)$). *At every step t , $\text{Inf}_S(t)^2 \leq 2\bar{D}_t$.*

By contrapositive, $\bar{D}_t \approx 0$ implies $\text{Inf}_S(t) \approx 0$: small asymmetry guarantees an inert position. The proof bounds $\text{Inf}_S(t)$ by total variation distance using Lemma 1, then applies Pinsker’s inequality (Appendix C.2).

4.3 Sign of \hat{D}_t Identifies Which Direction to Push

At a critical position, the sign of $\hat{D}_t(y_t)$ determines which way to push. Two regimes follow directly from the definition $\hat{D}_t(v) := \log P_S^t(v) - \log P_T^t(v)$:

- $\hat{D}_t(v) < 0$: the token v is more likely under the teacher ($P_T^t > P_S^t$), a choice the teacher would have predicted. Reinforcing such tokens follows the teacher’s path, the *exploit* direction.
- $\hat{D}_t(v) > 0$: conversely, v is a choice against the teacher’s prediction ($P_S^t > P_T^t$). Reinforcing such tokens moves the student onto a self-driven path consistent with success, the *explore* direction.

While the analysis above defines the teacher through the abstract event $R = 1$, this event cannot be conditioned on directly. In practice, we realize the teacher by feeding a known correct solution c as the conditioning context, so that $P_T^t(\cdot) = \pi_\theta(\cdot | h_t, c)$ serves as one instantiation of $\pi_\theta(\cdot | h_t, R = 1)$.

To verify that the sign of \hat{D}_t captures the explore/exploit direction, we ask which tokens the student systematically chooses against the teacher’s prediction versus which tokens align with it across rollouts from Qwen3-8B on DAPO-Math-17k [28]. We score each token’s polarization between the two sides with the smoothed log-odds z -score of Monroe et al. [16]. Figure 3 shows that explore-leaning tokens open new reasoning paths (*wait, another, consider*), while exploit-leaning tokens close them with verdicts and conclusions (*conclude, correct, final*). Full details of the marker selection and the per-category list are provided in Appendix D.

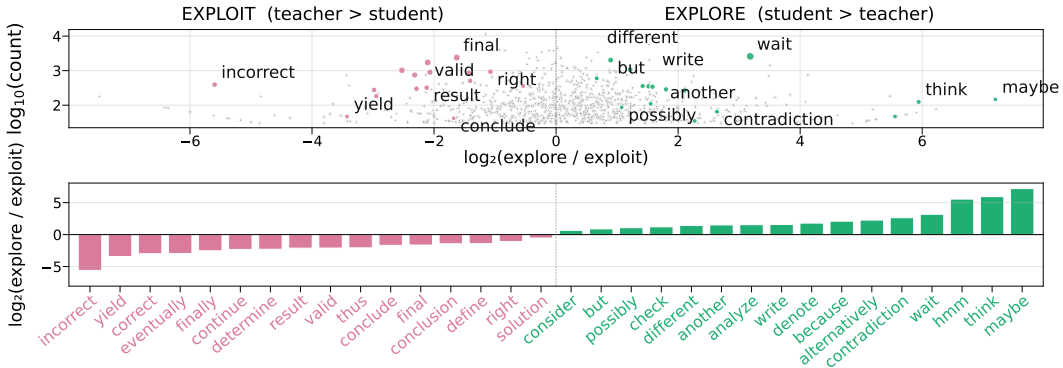


Figure 3: **Reasoning markers in the explore/exploit population.** (a) Volcano scatter of linguistic tokens: x -axis is the polarization $\log_2(n_{\text{explore}}/n_{\text{exploit}})$, y -axis is the total count $\log_{10}(n_{\text{explore}} + n_{\text{exploit}})$. Highlighted points (green = explore, pink = exploit) are categorized discourse markers; grey points are uncategorized tokens. (b) Per-marker polarization for these markers, sorted from most exploit-leaning (left) to most explore-leaning (right).

5 RLRT: RLVR with Reversed Teacher

We now present **RLRT** (RLVR with Reversed Teacher), an instance of the framework in Section 4 that uses an informed teacher and amplifies, on correct rollouts, tokens with $\hat{D}_t > 0$. RLRT modifies only

the token-level credit assignment of standard GRPO [19], leaving the rollout, reward, and trust-region machinery unchanged. Figure 4 provides a conceptual illustration and the training pipeline of RLRT.

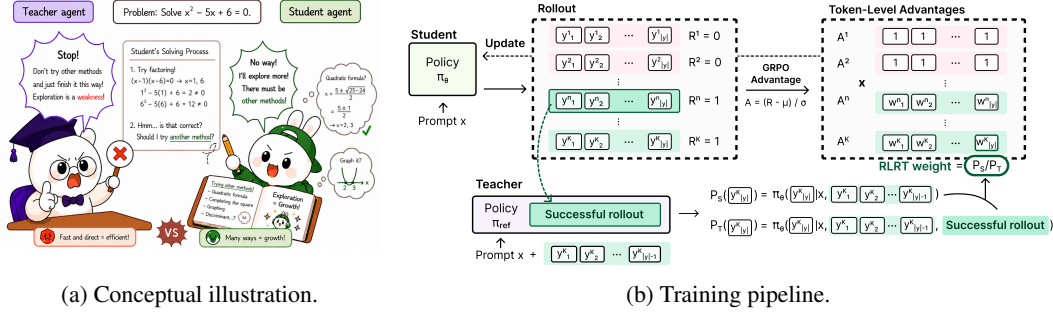


Figure 4: **Overview of RLRT.** (a) Conceptual illustration of the reversed-teacher signal. (b) Given a prompt x , the student policy π_θ generates K rollouts that receive verifiable rewards $r \in \{0, 1\}$ and group-standardized advantages $A^{(k)}$. A reversed teacher provides token-level signals \hat{D}_t that, on correct rollouts, up-weight tokens with $\hat{D}_t > 0$.

Reverse Weight as Token-Level Information Asymmetry Credit. For a prompt x , the student policy π_θ samples a group of K rollouts $\{y^{(k)}\}_{k=1}^K$, each receiving a verifiable reward $r(y^{(k)}) \in \{0, 1\}$ and a group-standardized advantage $A^{(k)}$. RLRT defines a per-token reweighting based on \hat{D}_t :

$$w_t^{\text{RLRT}} = \exp(\text{sign}(A) \cdot \hat{D}_t) = \left(\frac{P_S^t(y_t)}{P_T^t(y_t)} \right)^{\text{sign}(A)}. \quad (6)$$

On positive-advantage tokens, $w_t^{\text{RLRT}} > 1$ exactly when $\hat{D}_t > 0$, i.e., for tokens the student chose against the teacher’s prediction, and the reweighting amplifies these self-driven choices rather than aligning the student to the teacher. The flipping of the teacher/student ratio relative to the RLSD update [25] (Eq. 2) reflects a difference in intent: RLSD treats teacher–student disagreement as a correction to be applied, whereas RLRT treats it as a signal of valuable exploration and amplifies it.

Reward-Gated Update. Following the framework’s requirement that token-level information asymmetry be combined with outcome conditioning to target self-driven tokens on correct trajectories, the reverse weight is applied only to correct rollouts:

$$A_t^{\text{RLRT},(k)} = \begin{cases} A^{(k)} \cdot \left[(1 - \lambda) + \lambda \cdot \text{clip}(w_t^{\text{RLRT}}, 1 - \varepsilon_w, 1 + \varepsilon_w) \right] & \text{if } r(y^{(k)}) = 1, \\ A^{(k)} & \text{if } r(y^{(k)}) = 0, \end{cases} \quad (7)$$

where $\lambda \in [0, 1]$ controls the strength of the reversed signal ($\lambda = 0$ recovers vanilla GRPO, $\lambda = 1$ yields full reverse weighting), and the clip ε_w bounds the per-token advantage perturbation by $\lambda \cdot \varepsilon_w$.

6 Experiments

We design our experiments to verify that RLRT effectively leverages the information asymmetry signal to induce *valuable exploration* during RLVR training. Concretely, we ask:

- (Q1) How does RLRT, which pushes the student *away from* the teacher on correct rollouts, perform compared to self-distillation methods that pull the student *toward* the teacher?
- (Q2) Does \bar{D}_t causally identify critical positions, and does RLRT amplify their effect?
- (Q3) Beyond sharpening the base’s confident predictions, does RLRT introduce meaningful change?
- (Q4) Does RLRT induce more effective exploration than prior exploration-based methods?

6.1 Benchmark Results

Experimental Setup. To answer (Q1), we use DAPO-Math-17k [28] as the training corpus. Since post-training dynamics depend strongly on the pretrained checkpoint’s inductive biases [31, 30],

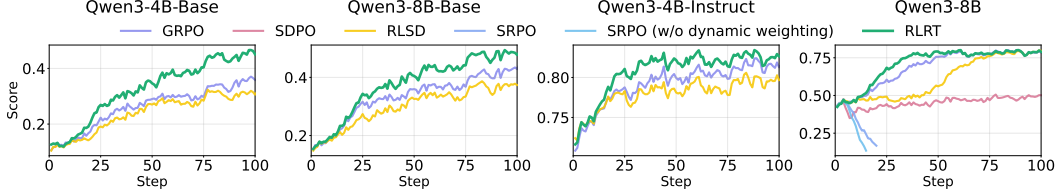


Figure 5: Training score curves across four backbones (Qwen3-4B-Base, Qwen3-8B-Base, Qwen3-4B-Instruct, Qwen3-8B). RLRT achieves faster exploration and higher training scores in all settings.

Table 1: Performance comparison across mathematical reasoning benchmarks. We report avg@16 and pass@16 for each benchmark. Δ denotes the gain of RLRT over the best of the other methods. Due to space constraints, results for Qwen3-4B-Instruct are in Table 3 (Appendix F.1).

Method	AIME24		AIME25		AIME26		HMMT26		AMC23		MATH500	
	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16
<i>Qwen3-4B-Base</i>	9.6	33.3	6.9	30.0	6.5	16.7	3.6	24.2	43.3	90.0	66.8	92.2
GRPO	15.0	40.0	14.4	33.3	12.3	36.7	10.0	27.3	58.3	87.5	80.2	94.2
RLSD	13.3	40.0	11.2	33.3	9.0	26.7	6.2	27.3	55.2	82.5	77.9	91.2
RLRT (Ours)	22.5	50.0	18.5	36.7	19.8	40.0	15.9	33.3	63.9	95.0	83.8	94.2
Δ vs. best	+7.5	+10.0	+4.1	+3.4	+7.5	+3.3	+5.9	+6.0	+5.6	+5.0	+3.6	0.0
<i>Qwen3-8B-Base</i>	10.4	33.3	10.2	30.3	9.8	30.0	5.3	30.3	56.3	85.0	74.4	93.0
GRPO	19.8	40.0	17.5	36.7	16.5	36.7	11.0	33.3	62.5	90.0	83.6	95.4
RLSD	17.3	40.0	15.0	33.3	13.5	36.7	8.1	27.3	64.2	87.5	80.9	92.4
RLRT (Ours)	27.9	63.3	18.8	50.0	21.9	53.3	15.9	33.3	67.3	97.5	84.4	95.6
Δ vs. best	+8.1	+23.3	+1.3	+13.3	+5.4	+16.6	+4.9	0.0	+3.1	+7.5	+0.8	+0.2
<i>Qwen3-8B (Thinking off)</i>	25.2	63.3	20.0	43.3	15.4	50.0	20.3	33.3	67.0	95.0	83.7	95.8
GRPO	70.2	86.7	59.4	83.3	62.9	86.7	41.7	66.7	93.6	100.0	94.8	98.2
SDPO	26.9	63.3	22.3	40.0	14.4	36.7	18.6	30.3	72.8	95.0	82.2	94.4
SRPO	15.4	26.7	9.8	26.7	8.3	26.7	9.7	21.2	49.2	77.5	75.0	90.2
RLSD	65.4	83.3	57.9	83.3	57.7	83.3	39.2	51.5	93.6	100.0	93.1	98.2
RLRT (Ours)	70.6	93.3	62.9	86.7	65.0	86.7	43.2	69.7	95.5	100.0	94.9	98.2
Δ vs. best	+0.4	+6.6	+3.5	+3.4	+2.1	0.0	+1.5	+3.0	+1.9	0.0	+0.1	0.0

we evaluate on three qualitatively distinct model types: a **base** model (Qwen3-4B/8B-Base), an **instruction-tuned** model (Qwen3-4B-Instruct), and a **thinking-tuned** model (Qwen3-8B).

We compare RLRT against GRPO and three self-distillation baselines, SDPO [9], SRPO [12], and RLSD [25]. We adopt SDPO rather than the closely related OPSD [32], since OPSD relies on ground-truth solutions from an external dataset and on a hybrid setup in which the student runs with thinking disabled and the teacher with thinking enabled. SDPO instead operates entirely on the model’s own rollouts, consistent with our self-distillation setup. Details of each algorithm are provided in Appendix G.1. In addition, SDPO collapsed early on Qwen3-4B/8B-Base (Appendix F.2), so we omit a detailed comparison for base models. We use a training batch size of 256, a PPO mini-batch size of 128, and a maximum response length of 20,480 tokens, with asymmetric clipping $\epsilon_{\text{high}}=0.28$ and $\epsilon_{\text{low}}=0.2$ following Yu et al. [28]. Further hyperparameters are listed in Appendix G.2.

Performance Comparison. Figure 5 shows the training curves for each algorithm, and Table 1 presents the evaluation results of the trained models on six math benchmarks using avg@16 and pass@16. As shown in Figure 5 and Table 1, across all four backbones, RLRT substantially outperforms both GRPO and the self-distillation baselines, exhibiting faster training-score growth and yielding significant average benchmark gains of **18.0%** (Qwen3-4B-Base), **12.0%** (Qwen3-8B-Base), **3.4%** (Qwen3-4B-Instruct), and **2.2%** (Qwen3-8B) over the baselines. Notably, SRPO, which routes correct rollouts to GRPO and incorrect rollouts to self-distillation, performs even worse than full self-distillation on math. We conjecture that self-distillation and GRPO promote different reasoning styles (e.g., exploration and exploitation as discussed in Section 4.3), leading to conflicting gradients. The gain is largest on Qwen3-4B-Base and smallest on Qwen3-8B, suggesting that RLRT’s exploration signal is most effective when the policy has not yet been concentrated by instruction tuning.

6.2 Causal Intervention via Reflection Injection

We answer (Q2) by injecting the reflection prompt “Wait, let me reconsider.” at a chosen token in a rollout and letting the model continue: if high- \bar{D}_t tokens are truly critical branch points, this should flip outcomes there more often than elsewhere. We run this on 100 DAPO-Math-17k problems (8 rollouts each) across Qwen3-8B checkpoints from step 0 (base) to step 100 under both RLRT and GRPO, injecting at three positions: $\arg \max_t \bar{D}_t$ (max_kl), a uniform-random token (random), and $\arg \min_t \bar{D}_t$ (min_kl). On the hard ($n_{\text{correct}} \in \{0, 1, 2\}$) and easy ($\{5, 6, 7\}$) subsets, we report $\text{flip} \rightarrow R$ (wrong \rightarrow right) and $\text{flip} \rightarrow W$ (right \rightarrow wrong) rates, respectively.

Two findings emerge from Fig. 6. First, on the untuned checkpoint (step 0, ■), $\text{flip} \rightarrow R$ at max_kl is twice that at random or min_kl, confirming Section 4.2’s claim that \bar{D}_t marks positions causally affecting correct outcomes. The absence of a comparable $\text{flip} \rightarrow W$ spike (panel b) reflects that the reflection prompt is biased toward correcting errors, though max_kl remains higher than random and min_kl. Second, the two algorithms diverge with training: RLRT amplifies the max_kl $\text{flip} \rightarrow R$ gain from $\sim 18\%$ to over 40% by step 100, while GRPO lets it collapse toward random and min_kl. RLRT’s $\text{flip} \rightarrow W$ declines just like GRPO’s, so these gains do not come at the cost of fragility on correct rollouts. This explains RLRT’s edge: its \bar{D}_t -weighted updates concentrate exploration credit on these critical positions, whereas GRPO spreads it across mostly inert tokens.

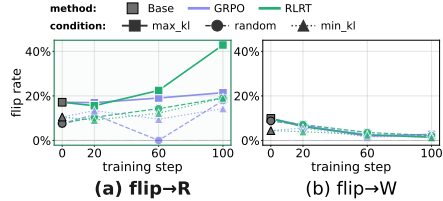


Figure 6: Reflection injected at max_kl (■), random (●), or min_kl (▲). (a) $\text{flip} \rightarrow R$ on hard subset; (b) $\text{flip} \rightarrow W$ on easy subset.

6.3 Does RLRT Lead to More Meaningful Distributional Shifts?

To answer (Q3), we analyze *where* and *how* each fine-tuned policy’s next-token distribution π_{ft} diverges from the base policy π_{base} , following Meng et al. [15]. We focus on hard prompts ($n_{\text{correct}} \in \{0, 1, 2\}$ out of 8 under π_{base}) so that any shift reflects how the policy learns to improve on cases the base struggles with, and use 30 such prompts from DAPO-Math-17k. At each token position along a fine-tuned rollout, we measure Jensen–Shannon divergence $\text{JS}(\pi_{\text{ft}} \parallel \pi_{\text{base}})$, and call positions with $\text{JS} > 0.1$ *high-divergence*: these are the tokens where π_{ft} has changed its mind relative to π_{base} .

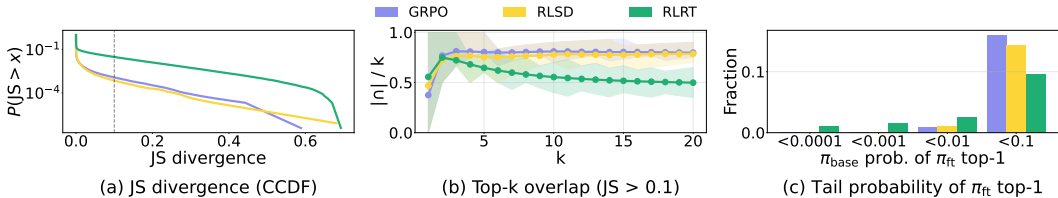


Figure 7: **Token-level distributional shifts of π_{ft} relative to π_{base} .** (a) CCDF of $\text{JS}(\pi_{\text{ft}} \parallel \pi_{\text{base}})$ across all positions; dashed line marks the $\text{JS} > 0.1$ threshold for (b)–(c). (b) Top- k overlap $|\text{top-}k(\pi_{\text{ft}}) \cap \text{top-}k(\pi_{\text{base}})|/k$ at high-divergence positions ($k \in [1, 20]$): how much the candidate set is reshuffled. (c) Fraction of high-divergence positions whose new top-1 token had π_{base} -probability below each threshold: how deep into the tail.

The three panels in Figure 7 answer three questions about the shift:

- **(a) How often does the policy diverge from the base?** Panel (a) shows the fraction of positions with JS divergence above threshold x . GRPO and RLSD stay close to π_{base} at most positions, while RLRT places far more positions in the high-divergence regime.
- **(b) When it diverges, do new tokens enter the top candidates, or are existing ones re-ranked?** Panel (b) measures top- k overlap between π_{ft} and π_{base} at high-divergence positions. GRPO and RLSD retain $\sim 80\%$ of π_{base} ’s candidates even at $k \geq 3$, re-weighting the existing pool. RLRT drops to $\sim 50\%$ at $k=20$, indicating many top candidates are tokens the base did not surface.
- **(c) How extreme are these new candidates?** Panel (c) reports the fraction of high-divergence positions whose new top-1 token had π_{base} -probability below each threshold. RLRT promotes tokens with base probability under 10^{-3} to top-1 over $10\times$ as often as the others, routinely picking tokens the base treated as essentially zero.

Together, the three views draw a clear line. GRPO and RLSD *sharpen* what π_{base} already prefers, re-weighting its top candidates. RLRT instead *reorganizes* the candidate set itself, pulling tokens from the base’s tail into top positions: it goes beyond reinforcing what the base knows and produces genuinely new behavior.

6.4 Comparison with Other Exploration Methods

We finally answer (Q4) by comparing RLRT against two representative exploration methods: GRPO with an entropy bonus (GRPO+EB) [3] for token-level entropy regulation, and DIVER [8] for sequence-level diversity.

For each method, we evaluate performance on Qwen3-8B-Base by comparing the pass@ k curve for $k \in \{1, 2, \dots, 256\}$ on AIME24 and AIME26. We sample 256 responses per problem and compute pass@ k using the unbiased estimator of Chen et al. [1]. As shown in Figure 8, GRPO+EB injects only local stochasticity at individual decision points [8] and tracks GRPO closely across the pass@ k curve, even falling below GRPO at small k . DIVER improves on GRPO, most visibly at large k , but its margin remains narrow, suggesting that its semantic-level diversity heuristic broadens exploration only modestly. RLRT, in contrast, dominates from pass@1 through pass@256, reflecting genuinely broader coverage across reasoning modes rather than within one.

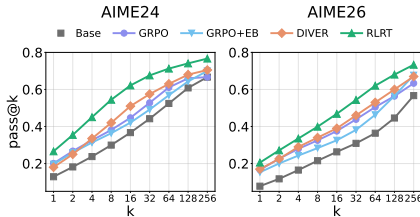


Figure 8: Pass@ k comparison across exploration methods on AIME24 and AIME26.

6.5 Ablation Study

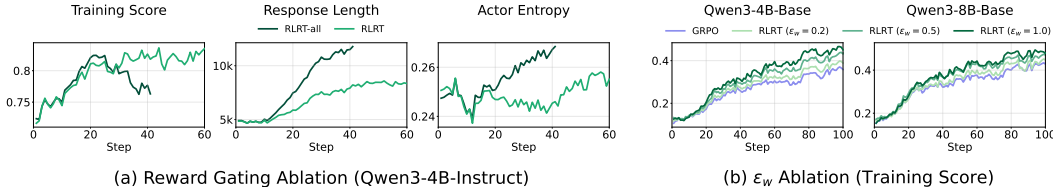


Figure 9: **RLRT ablations.** (a) reward gating on Qwen3-4B-Instruct: RLRT vs. RLRT-all (no $r=1$ gating) on training score, response length, and actor entropy. (b) clipping range ϵ_w on Qwen3-4B-Base and Qwen3-8B-Base, with GRPO as reference.

RLRT without Reward Gating. As described in Section 5, RLRT applies the reverse weight only on correct rollouts ($r=1$). To isolate the effect of the reward gate, we compare against **RLRT-all**, which applies the same weight regardless of correctness. As shown in Figure 9 (a), RLRT-all initially tracks RLRT but then diverges: response length and entropy grow unbounded, and training collapses around step 40. This confirms that RLRT’s gain requires restricting the reverse weight to correct rollouts: without the gate, the reverse weight reinforces teacher-divergent tokens on *failed* rollouts, conflating valuable exploration with spurious divergence.

Effect of the Clipping Range ϵ_w . The clipping range ϵ_w controls how strongly the reverse weight can deviate from 1, and thus how much it reshapes the gradient on correct rollouts. We sweep $\epsilon_w \in \{0.2, 0.5, 1.0\}$ on Qwen3-4B/8B-Base against a GRPO baseline. Figure 9 (b) shows that tighter clipping keeps the reverse weight near unity and tracks GRPO closely, while looser clipping ($\epsilon_w = 1.0$) yields the strongest training score on both backbones. This confirms that RLRT’s gains come from the reweighting itself, not from $r=1$ filtering alone.

7 Conclusion

We presented **RLRT**, which inverts self-distillation on correct rollouts: rather than pulling the student toward a privileged-context teacher, it amplifies tokens where the student diverged from the teacher yet still succeeded. We formalized self-driven reasoning through information asymmetry and

demonstrated its effectiveness as an exploration signal both theoretically and empirically. Experiments on base, instruction-tuned, and thinking-tuned Qwen3 yield substantial gains over GRPO, self-distillation, and exploration baselines. Extending RLRT to noisier rewards, other forms of asymmetry, and broader on-policy distillation beyond self-distillation, where the teacher distribution may come from diverse sources, is left for future work.

Acknowledgments

This work was supported by Microsoft Research and in part by grants from the Institute of Information & Communications Technology Planning & Evaluation (IITP), funded by the Korea government (MSIT), under Grant No. RS-2024-00457882 (AI Research Hub Project) and Grant No. RS-2022-II220469 (Development of Core Technologies for Task-oriented Reinforcement Learning for Commercialization of Autonomous Drones).

References

- [1] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.
- [2] Zhipeng Chen, Xiaobo Qin, Youbin Wu, Yue Ling, Qinghao Ye, Wayne Xin Zhao, and Guang Shi. Pass@ k training for adaptively balancing exploration and exploitation of large reasoning models. *arXiv preprint arXiv:2508.10751*, 2025.
- [3] Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30377–30385, 2026.
- [4] 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.
- [5] Shihan Dou, Muling Wu, Jingwen Xu, Rui Zheng, Tao Gui, and Qi Zhang. Improving rl exploration for llm reasoning through retrospective replay. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 594–606. Springer, 2025.
- [6] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-rl: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- [7] Zhezheng Hao, Hong Wang, Haoyang Liu, Jian Luo, Jiarui Yu, Hande Dong, Qiang Lin, Can Wang, and Jiawei Chen. Rethinking entropy interventions in rlvr: An entropy change perspective. *arXiv preprint arXiv:2510.10150*, 2025.
- [8] Zican Hu, Shilin Zhang, Yafu Li, Jianhao Yan, Xuyang Hu, Leyang Cui, Xiaoye Qu, Chunlin Chen, Yu Cheng, and Zhi Wang. Diversity-incentivized exploration for versatile reasoning. *arXiv preprint arXiv:2509.26209*, 2025.
- [9] Jonas Hübner, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Büning, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.
- [10] Renren Jin, Pengzhi Gao, Yuqi Ren, Zhuowen Han, Tongxuan Zhang, Wuwei Huang, Wei Liu, Jian Luan, and Deyi Xiong. Revisiting entropy in reinforcement learning for large reasoning models. *arXiv preprint arXiv:2511.05993*, 2025.
- [11] Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Why does self-distillation (sometimes) degrade the reasoning capability of llms? *arXiv preprint arXiv:2603.24472*, 2026.

- [12] Gengsheng Li, Tianyu Yang, Junfeng Fang, Mingyang Song, Mao Zheng, Haiyun Guo, Dan Zhang, Jinqiao Wang, and Tat-Seng Chua. Unifying group-relative and self-distillation policy optimization via sample routing. *arXiv preprint arXiv:2604.02288*, 2026.
- [13] Zeyuan Liu, Jeonghye Kim, Xufang Luo, Dongsheng Li, and Yuqing Yang. Exploratory memory-augmented LLM agent via hybrid on- and off-policy optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL <https://openreview.net/forum?id=U0zxviKVFO>.
- [14] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding rl-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025. URL <https://openreview.net/forum?id=5PAF7PAY2Y>.
- [15] Haoming Meng, Kexin Huang, Shaohang Wei, Chiyu Ma, Shuo Yang, Xue Wang, Guoyin Wang, Bolin Ding, and Jingren Zhou. Sparse but critical: A token-level analysis of distributional shifts in rlvr fine-tuning of llms. *arXiv preprint arXiv:2603.22446*, 2026.
- [16] Burt L Monroe, Michael P Colaresi, and Kevin M Quinn. Fightin’ words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403, 2008.
- [17] Phuc Minh Nguyen, Chinh D La, Duy MH Nguyen, Nitesh V Chawla, Binh T Nguyen, and Khoa D Doan. The reasoning boundary paradox: How reinforcement learning constrains language models. *arXiv preprint arXiv:2510.02230*, 2025.
- [18] Jaesung R Park, Junsu Kim, Gyeongman Kim, Jinyoung Jo, Sean Choi, Jaewoong Cho, and Ernest K Ryu. Clip-low increases entropy and clip-high decreases entropy in reinforcement learning of large language models. *arXiv preprint arXiv:2509.26114*, 2025.
- [19] 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.
- [20] Idan Shenfeld, Mehul Damani, Jonas Hübner, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.
- [21] Yuda Song, Julia Kempe, and Remi Munos. Outcome-based exploration for llm reasoning. *arXiv preprint arXiv:2509.06941*, 2025.
- [22] Yuda Song, Lili Chen, Fahim Tajwar, Remi Munos, Deepak Pathak, J Andrew Bagnell, Aarti Singh, and Andrea Zanette. Expanding the capabilities of reinforcement learning via text feedback. *arXiv preprint arXiv:2602.02482*, 2026.
- [23] Zhongwei Wan, Yun Shen, Zhihao Dou, Donghao Zhou, Yu Zhang, Xin Wang, Hui Shen, Jing Xiong, Chaofan Tao, Zixuan Zhong, et al. Dsdr: Dual-scale diversity regularization for exploration in llm reasoning. *arXiv preprint arXiv:2602.19895*, 2026.
- [24] Chen Wang, Zhaochun Li, Jionghao Bai, Yuzhi Zhang, Shisheng Cui, Zhou Zhao, and Yue Wang. Arbitrary entropy policy optimization breaks the exploration bottleneck of reinforcement learning. *arXiv preprint arXiv:2510.08141*, 2025.
- [25] Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026.
- [26] Xinhao Yao, Lu Yu, Xiaolin Hu, Fengwei Teng, Qing Cui, Jun Zhou, and Yong Liu. The debate on rlvr reasoning capability boundary: Shrinkage, expansion, or both? a two-stage dynamic view. *arXiv preprint arXiv:2510.04028*, 2025.
- [27] Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.
- [28] Qiyang 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.

- [29] 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? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL <https://openreview.net/forum?id=40sgYD7em5>.
- [30] Charlie Zhang, Graham Neubig, and Xiang Yue. On the interplay of pre-training, mid-training, and rl on reasoning language models. *arXiv preprint arXiv:2512.07783*, 2025.
- [31] Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. Echo chamber: RL post-training amplifies behaviors learned in pretraining. *arXiv preprint arXiv:2504.07912*, 2025.
- [32] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

A Limitations and Future Directions

To our knowledge, RLRT is the first to show that reversing the teacher’s signal, rather than aligning to it, can improve RLVR combined with distillation. We propose reading the information-asymmetric signal between teacher and student as a driver of exploration rather than imitation, and provide empirical evidence that this reinterpretation yields consistent gains across diverse model families. However, our setup is limited in two ways: it relies on a self-distillation framework where the teacher and student share parameters, and the experiments are restricted to mathematical reasoning.

RLRT opens several directions for future work. One axis is varying the teacher itself: rather than self-distillation, the teacher could be a separate, stronger reasoning model (as in on-policy distillation), or, conversely, a weaker one. A second axis is varying the form of privileged information given to the teacher, e.g., process-level feedback, partial hints, or failed attempts rather than a complete successful rollout. A third axis is characterizing how RLRT behaves under off-policy distillation, in contrast to the on-policy setting we study. A particularly promising direction across these axes is a hybrid that adaptively routes between teacher-guided and self-driven updates depending on the context.

B RLRT Algorithm

Algorithm 1 summarizes the full RLRT update. The only structural changes relative to GRPO are (i) the per-token reverse weight in Eq. (6) and (ii) the reward gate in Eq. (7); the rollout, reward, and trust-region mechanisms are otherwise unchanged.

Algorithm 1 RLRT: RLVR with Reversed Teacher

Require: Student/teacher π_θ ; privileged context c for the teacher; prompt x ; group size K ; mixing $\lambda \in [0, 1]$; clip radius ε_w .

- 1: Sample group $\{y^{(k)}\}_{k=1}^K \sim \pi_\theta(\cdot | x)$.
- 2: Compute verifiable reward $r(y^{(k)}) \in \{0, 1\}$ and group-standardized advantage $A^{(k)}$ for each k .
- 3: **for** each trajectory $k = 1, \dots, K$ **do**
- 4: **if** $r(y^{(k)}) = 1$ **then**
- 5: **for** $t = 1, \dots, |y^{(k)}|$ **do**
- 6: Compute $\hat{D}_t = \log P_S^t(y_t^{(k)}) - \log P_T^t(y_t^{(k)})$ {token-level information asymmetry}
- 7: $w_t^{\text{RLRT}} \leftarrow \exp(\text{sign}(A^{(k)}) \cdot \hat{D}_t)$ {Eq. (6)}
- 8: $A_t^{\text{RLRT},(k)} \leftarrow A^{(k)} \cdot \left[(1 - \lambda) + \lambda \cdot \text{clip}(w_t^{\text{RLRT}}, 1 - \varepsilon_w, 1 + \varepsilon_w) \right]$ {Eq. (7)}
- 9: **end for**
- 10: **else**
- 11: $A_t^{\text{RLRT},(k)} \leftarrow A^{(k)}$ for all t {vanilla GRPO advantage}
- 12: **end if**
- 13: **end for**
- 14: Update θ with the standard GRPO surrogate using $\{A_t^{\text{RLRT},(k)}\}$.

C Proofs and Supporting Results

C.1 Proof of Lemma 1

By the definition of P_T^t and Bayes' rule,

$$P_T^t(v) = \pi_\theta(v | h_t, R = 1) = \frac{\pi_\theta(R = 1 | h_t, y_t = v) \cdot \pi_\theta(v | h_t)}{\pi_\theta(R = 1 | h_t)}.$$

The numerator and denominator simplify using the definitions

$$f(v) := \Pr[R = 1 | h_t, y_t = v], \quad \bar{f}_S^t := \mathbb{E}_{v \sim P_S^t}[f(v)] = \Pr[R = 1 | h_t],$$

together with $\pi_\theta(v | h_t) = P_S^t(v)$, yielding

$$P_T^t(v) = \frac{f(v) \cdot P_S^t(v)}{\bar{f}_S^t}.$$

Taking logarithms,

$$\log P_T^t(v) = \log f(v) + \log P_S^t(v) - \log \bar{f}_S^t.$$

Applying the definition $\hat{D}_t(v) := \log P_S^t(v) - \log P_T^t(v)$ gives

$$\hat{D}_t(v) = \log \bar{f}_S^t - \log f(v).$$

□

C.2 Proof of Theorem 2

The proof has two steps: Step 1 expresses $\text{Inf}_S(t)$ in closed form using Lemma 1 and bounds it by total variation distance; Step 2 applies Pinsker's inequality.

Step 1: bound by total variation. By Lemma 1, $f(v) = \bar{f}_S^t \cdot P_T^t(v) / P_S^t(v)$, hence

$$f(v) - \bar{f}_S^t = \bar{f}_S^t \left(\frac{P_T^t(v)}{P_S^t(v)} - 1 \right) = \frac{\bar{f}_S^t}{P_S^t(v)} (P_T^t(v) - P_S^t(v)).$$

Substituting into the definition of $\text{Inf}_S(t)$,

$$\begin{aligned} \text{Inf}_S(t) &= \mathbb{E}_{v \sim P_S^t}[|f(v) - \bar{f}_S^t|] \\ &= \sum_{v \in \mathcal{V}} P_S^t(v) \cdot \frac{\bar{f}_S^t}{P_S^t(v)} |P_T^t(v) - P_S^t(v)| \\ &= \bar{f}_S^t \sum_{v \in \mathcal{V}} |P_T^t(v) - P_S^t(v)| \\ &= 2 \bar{f}_S^t \cdot \text{TV}(P_S^t, P_T^t) \\ &\leq 2 \text{TV}(P_S^t, P_T^t), \end{aligned}$$

where the second-to-last equality uses the definition $\text{TV}(P, Q) := \frac{1}{2} \sum_v |P(v) - Q(v)|$, and the last inequality uses $\bar{f}_S^t \in [0, 1]$.

Step 2: Pinsker's inequality. For any two probability distributions P, Q on a common space, Pinsker's inequality states

$$\text{TV}(P, Q) \leq \sqrt{\frac{1}{2} \text{KL}(P \| Q)}.$$

Applied to $P = P_S^t$ and $Q = P_T^t$,

$$\text{TV}(P_S^t, P_T^t) \leq \sqrt{\frac{1}{2} \bar{D}_t}.$$

Squaring the inequality from Step 1 and applying this bound,

$$\text{Inf}_S(t)^2 \leq 4 \text{TV}(P_S^t, P_T^t)^2 \leq 4 \cdot \frac{1}{2} \bar{D}_t = 2 \bar{D}_t,$$

as claimed. □

D Marker Statistics for the Explore/Exploit Reading

Starting from 8 rollouts of Qwen3-8B on each of 100 DAPO-Math-17k problems, we retain one correct and one incorrect trajectory per problem (200 trajectories total). At every position t of each trajectory, we identify two tokens from the entire vocabulary \mathcal{V} : $\arg \max_{v \in \mathcal{V}} \hat{D}_t(v)$ (the token most favored by the student over the teacher) is added to the explore corpus, and $\arg \min_{v \in \mathcal{V}} \hat{D}_t(v)$ (the token most favored by the teacher over the student) is added to the exploit corpus. Note that these are *not* the sampled tokens y_t ; they are the vocabulary entries where the student–teacher divergence is most extreme in each direction. We score every token type v that appears at least 30 times across the two corpora combined, after restricting to ASCII alphabetic tokens of length 3 to 15 characters. Let e_v, x_v be the counts of token v in the explore/exploit corpora (totals E, X). We compute polarization with the smoothed log-odds z -score of Monroe et al. [16],

$$z_v = \frac{\delta_v}{\sqrt{\widehat{\text{Var}}(\delta_v)}}, \quad \delta_v = \log \frac{e_v + \alpha}{E - e_v + \alpha} - \log \frac{x_v + \alpha}{X - x_v + \alpha}, \quad (8)$$

with $\alpha = 0.5$, where $z_v \gg 0$ marks reliable explore tokens and $z_v \ll 0$ marks reliable exploit tokens. We keep tokens with $|z_v| \geq 3$ (251 explore-side and 171 exploit-side candidates), then remove stopwords using two lists: NLTK English stopwords (198 words) and a domain-specific list (approximately 400 words covering math vocabulary, Greek letters, LaTeX fragments, tokenizer artifacts, English numerals, and generic non-discourse fillers). This yields **38** explore-side and **61** exploit-side markers, all listed in Table 2 with their z_v values, ranked by $|z_v|$ within each category.

Table 2: All explore/exploit markers retained after $|z_v| \geq 3$ filtering and automatic stoplist removal, grouped by discourse function. z_v is the smoothed log-odds polarization score (positive = teacher-suppressed; negative = teacher-favored). The “Other” rows list all tokens not mapping to any predefined category (13 explore, 32 exploit).

Direction	Category	Tokens (z_v)
Explore	Reflection	<i>wait</i> (33.6), <i>back</i> (4.8), <i>hmm</i> (3.2)
	Deliberation verb	<i>let</i> (35.1), <i>denote</i> (8.8), <i>write</i> (8.8), <i>note</i> (8.4), <i>think</i> (6.1), <i>consider</i> (5.5), <i>look</i> (5.5), <i>analyze</i> (4.9)
	Metacognitive verb	<i>check</i> (12.6), <i>recall</i> (7.7), <i>see</i> (6.1)
	Alternative marker	<i>another</i> (8.6), <i>different</i> (8.3), <i>alternative</i> (4.0), <i>alternatively</i> (3.5)
	Logical connective	<i>therefore</i> (7.0), <i>via</i> (5.3), <i>similarly</i> (4.7)
	Contrastive	<i>still</i> (13.1)
	Epistemic hedge	<i>maybe</i> (4.0), <i>perhaps</i> (3.3)
	Causal pivot	<i>since</i> (3.6)
	Other	<i>answer</i> (9.3), <i>icky</i> (6.7), <i>ones</i> (5.2), <i>problem</i> (5.2), <i>contradiction</i> (5.1), <i>pick</i> (4.4), <i>passes</i> (4.4), <i>split</i> (4.0), <i>infected</i> (3.9), <i>rolling</i> (3.9), <i>hat</i> (3.8), <i>sided</i> (3.8), <i>cover</i> (3.7)
Exploit	Decision verb	<i>determine</i> (−10.3), <i>define</i> (−9.8), <i>apply</i> (−5.3), <i>ensure</i> (−4.9), <i>means</i> (−4.8), <i>verify</i> (−4.7), <i>yield</i> (−4.6), <i>evaluate</i> (−4.4), <i>compute</i> (−3.8), <i>conclude</i> (−3.2)
	Verdict adjective	<i>valid</i> (−23.7), <i>incorrect</i> (−11.0), <i>correct</i> (−10.9), <i>right</i> (−10.6), <i>invalid</i> (−8.1)
	Consequential conn.	<i>thus</i> (−17.0), <i>however</i> (−6.2), <i>though</i> (−5.0), <i>although</i> (−3.6)
	Finalization	<i>final</i> (−23.8), <i>finally</i> (−19.9), <i>eventually</i> (−8.8)
	Outcome noun	<i>conclusion</i> (−12.9), <i>result</i> (−10.3), <i>results</i> (−5.2)
	Implication	<i>leads</i> (−5.0), <i>implies</i> (−3.1)
	Progression	<i>continue</i> (−16.4)
	Specification	<i>defined</i> (−4.7)
	Other	<i>higher</i> (−16.1), <i>general</i> (−9.7), <i>left</i> (−9.5), <i>reverse</i> (−8.2), <i>directly</i> (−7.8), <i>always</i> (−7.0), <i>instead</i> (−6.6), <i>specific</i> (−5.6), <i>text</i> (−5.5), <i>tool</i> (−4.9), <i>able</i> (−4.5), <i>teams</i> (−4.3), <i>following</i> (−4.1), <i>guarantee</i> (−4.1), <i>repetition</i> (−3.9), <i>avoid</i> (−3.9), <i>ants</i> (−3.8), <i>cannot</i> (−3.7), <i>trial</i> (−3.7), <i>least</i> (−3.6), <i>constraint</i> (−3.6), <i>containing</i> (−3.6), <i>actually</i> (−3.6), <i>fully</i> (−3.5), <i>performance</i> (−3.4), <i>guaranteed</i> (−3.3), <i>evenly</i> (−3.3), <i>meaning</i> (−3.2), <i>specifically</i> (−3.1), <i>analysis</i> (−3.1), <i>appear</i> (−3.1), <i>without</i> (−3.1)

Varying the threshold $|z_v| \in \{2, 3, 5\}$ does not change the qualitative picture: the same discourse categories dominate on each side, and only the depth of each category’s tail changes.

E Further Examples of Critical Positions and Explore/Exploit Directions

Figure 2 in Section 4.1 illustrates the explore/exploit decomposition on a single trajectory. To show that this pattern is not an artifact of one example, Figure 10 presents an additional rollout annotated with the same \bar{D}_t heatmap and top-candidate display. The qualitative picture replicates: most tokens carry small \bar{D}_t , while a few high-asymmetry tokens mark critical positions. At these positions, exploit-leaning candidates ($\hat{D}_t < 0$, e.g., *Final*, *Conclusion*) push toward closing the argument, whereas explore-leaning candidates ($\hat{D}_t > 0$, e.g., *Can*, *Each*, *But*, *How*) open alternative reasoning paths the teacher would not have predicted. This consistency supports the use of $\text{sign}(\hat{D}_t)$ as a stable indicator of self-driven versus teacher-aligned tokens, as claimed in Section 4.3.

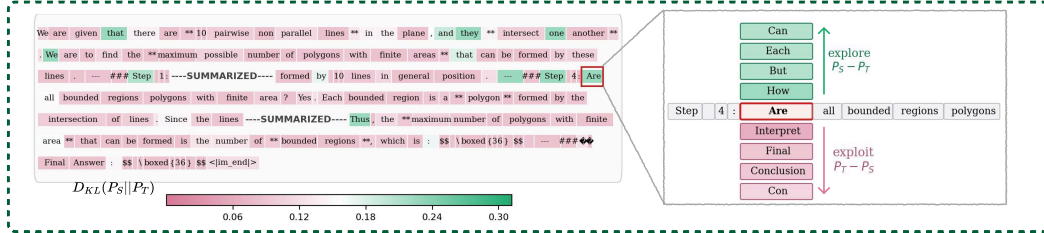


Figure 10: **Additional example of critical positions and explore/exploit directions**, complementing Figure 2. Token shading shows the position-level asymmetry $\bar{D}_t = \text{KL}(P_S^t \parallel P_T^t)$. At the highlighted critical position, candidates are taken as the union of the teacher’s and student’s top-100 tokens; we display the top four with the largest $P_S^t - P_T^t$ (green, $\hat{D}_t > 0$, *explore*) and the top four with the largest $P_T^t - P_S^t$ (pink, $\hat{D}_t < 0$, *exploit*).

F More Results

F.1 Benchmark Results on Qwen3-4B-Instruct

Extending Table 1, Table 3 reports results on Qwen3-4B-Instruct across six math benchmarks (AIME24/25/26, HMMT26, AMC23, and MATH500). Consistent with the discussion in Section 6.1, RLRT also achieves higher scores than other baselines on the Instruct-tuned model, yielding a 3.4% average improvement on avg@16 over the best baseline.

Table 3: Performance comparison on Qwen3-4B-Instruct across mathematical reasoning benchmarks. We report avg@16 and pass@16 for each benchmark. Δ denotes the gain of RLRT over the best of the other methods.

Method	AIME24		AIME25		AIME26		HMMT26		AMC23		MATH500	
	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16	Avg@16	Pass@16
<i>Qwen3-4B-Instruct</i>	64.4	86.7	47.3	76.7	54.0	83.3	36.4	57.6	94.7	100.0	94.1	97.8
GRPO	69.4	93.3	58.3	80.0	62.7	83.3	37.9	60.6	96.2	100.0	94.4	97.8
SDPO	53.1	83.3	37.1	70.0	42.9	73.3	31.6	45.5	89.1	100.0	91.8	97.4
RLSD	61.5	90.0	51.9	76.7	58.1	83.3	37.1	51.5	93.1	97.5	94.3	97.2
RLRT (Ours)	70.4	90.0	62.9	83.3	67.9	83.3	40.2	66.7	97.0	100.0	94.8	97.8
Δ vs. best	+1.0	-3.3	+4.6	+3.3	+5.2	0.0	+2.3	+6.1	+0.8	0.0	+0.4	0.0

F.2 Behavior of SDPO on Base Models

SDPO is a self-distillation method that uses the same model as both teacher and student under different conditioning contexts, and rapidly improves in-domain performance and induces more efficient reasoning by shortening response length [9]. However, it can become unstable in math reasoning due to its excessive suppression of hedging and reflective tokens (e.g., “wait”, “hmm”); these tokens are critical for robust reasoning [11]. We observe that this collapse is particularly severe on the Base model, where the score rapidly drops to 0 within 20 steps and the response length diverges compared to GRPO. We therefore exclude SDPO and its variant SRPO from the Base model comparison in our main table.



Figure 11: Training reward (left) and response length (right) on Qwen3-8B-Base. SDPO collapses quickly: its reward drops while response length blows up.

G Experimental Details

We build on the implementation of Kim et al. [11] (<https://github.com/beanie00/self-distillation-analysis>) and additionally implement GRPO with entropy bonus, SRPO, RLSD, and RLRT for our experiments. For DIVER [8], we use the official code (<https://github.com/NJU-RL/DIVER>) and train on the same DAPO-Math-17k corpus and hyperparameters as the other baselines. We run all experiments on 2×B200 GPUs. Training Qwen3-4B/8B-Base takes approximately one day, whereas Qwen3-4B-Instruct and Qwen3-8B require 2–3 days.

G.1 Details of Baseline Algorithms

All baselines share the GRPO surrogate and differ only in (i) the privileged context c defining the teacher view $P_T^t(\cdot) := \pi_\theta(\cdot | h_t, c)$, (ii) the per-token weight w_t on the advantage, and (iii) the trajectory-level gate. We write $\Delta_t := \text{sg}(\log P_T^t(y_t) - \log P_S^t(y_t))$ and $\hat{D}_t = -\Delta_t$ (Sec. 3).

- **GRPO** [19, 28]. The DAPO recipe: clip-higher ($\varepsilon_{\text{low}}=0.2$, $\varepsilon_{\text{high}}=0.28$), token-level loss aggregation, no KL penalty. No teacher view.
- **SDPO** [9]. Teacher conditions on a correct rollout; a logit-level KL loss pulls $P_S^t \rightarrow P_T^t$ on all rollouts.
- **SRPO** [12]. Same teacher as SDPO, but *routed by correctness*: SDPO loss on $r=0$ rollouts, GRPO on $r=1$, with entropy-aware dynamic weighting.
- **RLSD** [25]. Teacher conditions on the ground-truth answer. The reward fixes the update *direction*, while the teacher modulates only *magnitude*: $w_t^{\text{RLSD}} = (P_T^t(y_t)/P_S^t(y_t))^{\text{sign}(A)}$, applied to all rollouts.

We run each baseline with the primary settings recommended in its original paper.

Relation to RLSD. RLRT and RLSD use weights of the same form with *opposite exponents*, $w_t^{\text{RLRT}} = 1/w_t^{\text{RLSD}}$, and RLRT additionally gates on $r=1$. On correct rollouts, RLSD up-weights teacher-favored tokens ($\hat{D}_t < 0$); RLRT up-weights student-favored ones ($\hat{D}_t > 0$), amplifying self-driven reasoning rather than imitating the teacher.

Table 4: Baselines unified under the GRPO surrogate. Each method applies $A_t^{(k)} = A^{(k)} \cdot [(1 - \lambda) + \lambda \cdot \text{clip}(w_t, 1 - \varepsilon_w, 1 + \varepsilon_w)]$ and differs only in c , w_t , and the gate.

Method	Context c	Per-token weight w_t	Gate	Direction
GRPO	—	1	—	—
SDPO	correct rollout	logit-level KL: $\text{KL}(P_T^t P_S^t)$	all	$P_S \rightarrow P_T$
SRPO	correct rollout	SDPO loss if $r=0$; 1 if $r=1$	route by r	$P_S \rightarrow P_T$ on $r=0$
RLSD	ground truth	$(P_T^t(y_t)/P_S^t(y_t))^{\text{sign}(A)}$	all	teacher = magnitude
RLRT (ours)	correct rollout	$(P_S^t(y_t)/P_T^t(y_t))^{\text{sign}(A)}$	$r=1$ only	amplify self-driven

G.2 Hyperparameters

Training Hyperparameters. The training hyperparameters are listed in Table 5. For SDPO and SRPO, we follow the hyperparameter settings recommended in their original papers, sweeping only SRPO’s entropy-aware dynamic-weight coefficient $\beta \in \{0, 0.5, 1\}$ per model. For RLSD and RLRT, we share $\lambda_{\text{init}} = 0.5$ and sweep $\varepsilon_w \in \{0.2, 0.5, 1.0\}$ under an identical protocol. RLSD was consistently best with $\varepsilon_w = 0.2$, with larger values degrading performance below GRPO. RLRT, by contrast, remained above GRPO across the entire sweep. The best setting shifted modestly with the base model’s ability to explore diverse solution paths (base: 1.0, instruction-tuned: 0.5, thinking-tuned: 0.2), consistent with the role of ε_w in the method.

For GRPO, we follow Liu et al. [14] and disable std normalization of the advantage to preserve relative signal strength across groups. For RLSD and RLRT, we retain it following RLSD [25].

Table 5: Hyperparameters for GRPO, SDPO, SRPO, RLSD, and RLRT.

Category	Parameter	Value
<i>Common (shared by all methods)</i>		
Data	Max. prompt length	2048
	Max. response length	20480
Batching	Question batch size	256
	Mini batch size	128
	Number of rollouts	8
Rollout	Inference engine	vllm
	Temperature	1.0
Training	Optimizer	AdamW
	Warmup steps	10
	Weight decay	0.01
	Gradient clip norm	1.0
Policy loss	ϵ -low	0.2
	ϵ -high	0.28
	Loss aggregation	token-level
Advantage std normalization	GRPO, SRPO RLSD, RLRT	disabled enabled (Following RLSD [25])
Off-policy correction	Rollout IS clip	2
	KL coefficient (λ)	0.0
Learning rate	GRPO / RLSD / RLRT	1×10^{-6}
	SDPO	1×10^{-5}
	SRPO	5×10^{-6}
<i>SDPO / SRPO</i>		
Distillation	Divergence	Jensen-Shannon ($\alpha = 0.5$)
	Top- K distillation	100
	EMA update rate	0.0
	Entropy-aware coefficient (β , SRPO only)	swept over $\{0, 0.5, 1\}$
<i>RLSD / RLRT</i>		
Token reweighting	Initial mixing (λ_{init})	0.5
	ϵ_w sweep	$\{0.2, 0.5, 1.0\}$
	- Best (RLSD)	0.2
	- Best (RLRT)	1.0 (base), 0.5 (instruct), 0.2 (thinking)
	Mixing decay steps (RLSD)	50
	Mixing decay steps (RLRT)	no decay (base), 30 (instruct, thinking)

Evaluation Hyperparameters. Following the evaluation recommendations for each model², we use a maximum response length of 38912 tokens, temperature 0.7, top- $p = 0.8$, and top- $K = 20$ across all models.

²<https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507>, <https://huggingface.co/Qwen/Qwen3-8B>

H Full-Trajectory Heatmaps of \bar{D}_t

The figure in Section 4 highlights a single critical position per rollout. For completeness, we provide full-trajectory heatmaps of the position-level information asymmetry $\bar{D}_t = \text{KL}(P_S^t || P_T^t)$ across entire rollouts. Each token is shaded by its \bar{D}_t value: **greener** tokens are critical (token choice can change correctness), while **pinkier** stretches are routine. The heatmaps reveal two qualitative properties of the signal that the zoomed-in view cannot convey: (i) critical positions are sparse and concentrated, with the bulk of any rollout consisting of decision-insensitive tokens, and (ii) they cluster at semantically meaningful junctions such as step transitions, choice of solution strategy, and arithmetic commitments, rather than scatter uniformly.

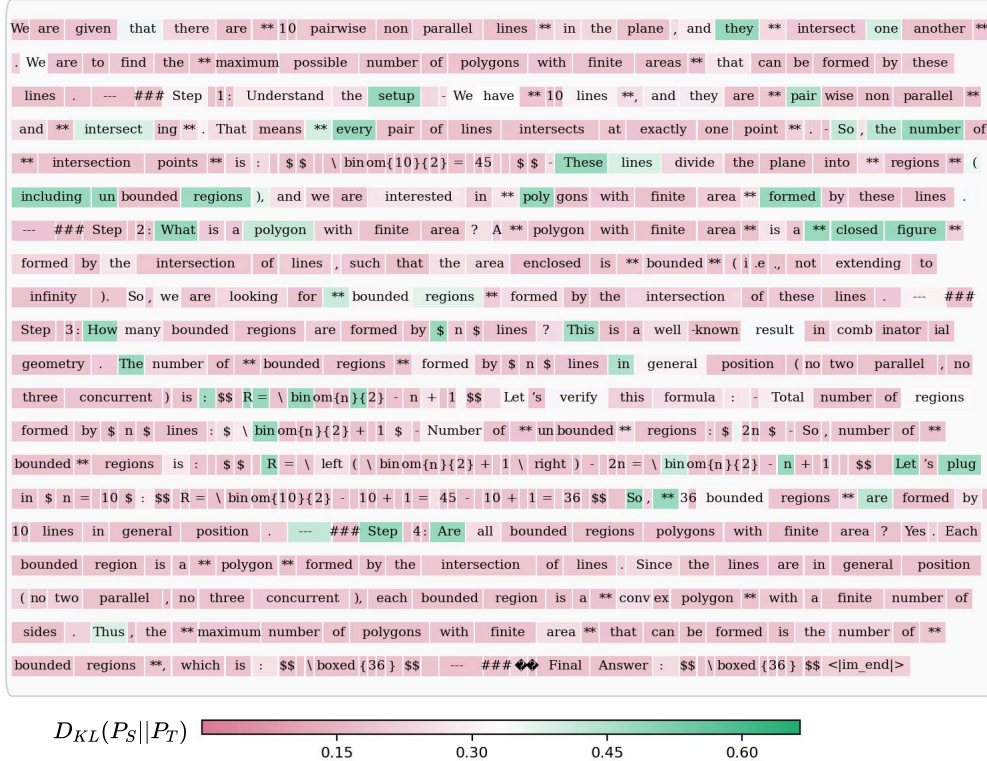


Figure 12: Full-trajectory heatmap of \bar{D}_t on the first example rollout. Critical positions (**green**) are sparse and concentrate at decision points, while long routine stretches (**pink**) carry little signal.

We are given that a state issues license plates consisting of **six digits**, each from 0 to 9. The requirement is that **any two license plates must differ in at least two places**. We are to determine the maximum number of distinct license plates that can be issued under this constraint. --- **Step 1: Understanding the Problem** Each license plate is a **6-digit string**, where each digit is from 0 to 9. So, the total number of possible license plates (without any restrictions) is: $10^6 = 1,000,000$. However, the state has a **minimum Hamming distance requirement** of **2** between any two license plates. That is, any two license plates must differ in **at least two positions**. We are to find the **maximum number of such license plates** that can be used under this constraint. --- **Step 2: Reformulate the Problem** This is a **coding theory** problem. Specifically, we are trying to find the **maximum size of a code** (a set of binary strings of length 6) with the following properties: Each "symbol" is a digit from 0 to 9 (so, not binary, but base-10). The **Hamming distance** between any two codewords is at least 2. This is equivalent to finding the **maximum size of a code** of length 6 over an alphabet of size 10, with **minimum Hamming distance 2**. Let's denote this maximum number as $A(6, 2, 10)$, where: 6 is the length of the code, 2 is the minimum Hamming distance, 10 is the size of the alphabet. --- **Step 3: Use the Sphere-Packing Bound** We can use the **sphere-packing bound** to estimate the maximum number of codewords. For a code of length n , over an alphabet of size q , with minimum Hamming distance d , the **sphere-packing bound** is: $A(n, d, q) \leq \frac{q^n}{\sum_{k=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{k} (q-1)^k}$ where $t = \lfloor \frac{d-1}{2} \rfloor$. In our case: $n = 6, d = 2$, so $t = \lfloor \frac{2-1}{2} \rfloor = 0$; $q = 10$. So the sphere-packing bound becomes: $A(6, 2, 10) \leq \frac{10^6}{\binom{6}{0} (10-1)^0} = \frac{1,000,000}{1} = 1,000,000$. So the sphere-packing bound gives us that the maximum number of license plates is **at most 1,000,000**, which is the total number of possible license plates. But this is **not useful** because it's not tight for this case. --- **Step 4: Use the Simple Bound from the Minimum Distance** We are looking for the **maximum number of codewords** in a code of length 6 over an alphabet of size 10, with **minimum Hamming distance 2**. Let's think about the following: If we fix one license plate, say "000000", then any other license plate must differ in **at least two digits**. So, how many license plates are **at Hamming distance less than 2** from "000000"? These are: **All license plates that are equal to "000000"** (1 plate) - All license plates that differ in **exactly 1 digit** (there are $6 \times 9 = 54$ such plates). So, the **number of plates that are not allowed** to be used if we include "000000" is: $1 + 54 = 55$. So, if we include "000000", then we can only include license plates that are **at least 2 digits away** from it. So, the **maximum number of license plates** is at most: $\frac{10^6}{55} = \frac{1,000,000}{55} \approx 18,181.81$. So, the **upper bound** is **18,181** (rounded down). This is the **sphere-packing bound**, and it gives us a **tight upper bound** for this case. --- **Step 5: Is This Bound Achievable?** Now we ask: **Can we actually construct a code of length 6 over an alphabet of size 10 with minimum Hamming distance 2, and size 18,181?** This is equivalent to asking: **Can we partition the set of all 10^6 license plates into groups (spheres) of size 55, such that each group contains all license plates within Hamming distance 1 of a codeword, and no two spheres overlap?** This is the **sphere-packing idea**. If we can do this, then we can achieve the bound. However, **in general**, it is **not always possible** to achieve the sphere-packing bound. But in this case, the **minimum distance is 2**, and the **sphere size is 55**, which divides evenly into 10^6 . Indeed: $10^6 \div 55 = 18,181.818 \dots$. But we can't have a fraction of a license plate. However, **if we take the floor**, we get: $\lfloor \frac{10^6}{55} \rfloor = 18,181$. So, the **maximum number of license plates** that can be used is: $\boxed{18,181}$. --- **Step 6: Final Answer** Thus, the **maximum number of distinct license plates** that the state can use, such that any two differ in at least two places, is: $\boxed{18181}$. --- **Final Answer**: $\boxed{18181}$.

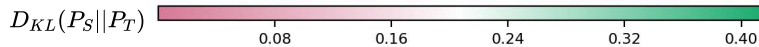


Figure 13: Full-trajectory heatmap of \bar{D}_t on the second example rollout (same conventions as Figure 12).