ON-POLICY DISTILLATION WITH VERIFIABLE REWARD

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning with Verifiable Rewards (RLVR) and on-policy distillation (OPD) have become two widely adopted paradigms for post-training large language models. However, RLVR suffers from sparse task-level feedback, while OPD provides dense token-level guidance but ignores trajectory correctness, limiting its performance to that of the teacher. Combining them is a promising direction: OPD supplies dense supervisory signals, while RLVR provides task-level correctness. Nevertheless, existing integrations often rely on weighted combination or heuristic switching, introducing extra hyperparameters and trade-offs. We propose **On-policy Distillation with Verifiable Reward (OPDVR)**, a simple yet effective method that seamlessly combines OPD and RLVR without adding any hyperparameters. We first reformulate the implicit reward of sampled-token OPD based on trajectory correctness, then apply a ReLU gating mechanism to ensure that correct trajectories receive non-negative rewards and incorrect ones receive non-positive rewards—thereby aligning the distillation signal with task success while preserving the teacher's distributional guidance. Furthermore, our modification transforms sampled-token OPD into a proper RLVR method, making it readily combinable with any policy gradient algorithm, such as GRPO. Experiments on six reasoning benchmarks show that OPDVR consistently outperforms OPD and recent methods.

1 INTRODUCTION

Reinforcement Learning with Verifiable Rewards (RLVR) has become a prevalent and effective post-training paradigm for reasoning tasks (Guo et al., 2025; Shao et al., 2024; Jaech et al., 2024; Trinh et al., 2024; Yang et al., 2024). RLVR delivers explicit reward signals based on task outcomes, such as correct mathematical final answers or compilable code, which directly align model optimization with core task objectives. However, these rewards are typically sparse, providing little supervision for intermediate steps and making credit assignment challenging. Recently, On-policy Distillation (OPD) has emerged as a promising post-training paradigm (Xu et al., 2026a; Xiao et al., 2026; Yang et al., 2025a; Zeng et al., 2026). Unlike RLVR, which relies on sparse outcome-level rewards, OPD leverages a teacher model to provide dense token-level guidance. However, OPD's objective is purely distributional—it drives the student to mimic the teacher's output distribution without considering whether the generated response is correct or incorrect, limiting the student's performance to that of the teacher.

The complementary limitations of RLVR and OPD make combining them a natural and promising direction, where OPD provides fine-grained dense guidance and RLVR offers reliable task-level correctness supervision. Existing approaches typically treat OPD and RLVR as two separate methods, either combining them explicitly or selectively applying one based on heuristic criteria. Despite their empirical effectiveness, such designs often introduce extra hyperparameters and heuristic trade-offs (Cai et al., 2026; Yang et al., 2026a; Li et al., 2026a; Wang et al., 2026).

In this work, we propose **On-policy Distillation with Verifiable Reward (OPDVR)**. We apply an extremely simple ReLU gating mechanism to sampled-token OPD, seamlessly combining OPD and RLVR without introducing any hyperparameters. We first revisit sampled-token OPD from an RLVR perspective and provide a mathematical reformulation of sampled-token OPD's reward. From the RLVR view, if we separate by trajectory correctness, sampled-token OPD can be viewed as applying token-level supervision by weighting binary task correctness rewards with teacher-student distribution discrepancies. Specifically, for sampled tokens that constitute correct reasoning trajectories, the model is assigned a token-level reward $+1$ weighted by $\log(\pi_T/\pi_\theta)$; for tokens leading to incorrect

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

trajectories, the model receives a reward -1 weighted by $\log(\pi_\theta/\pi_T)$. However, we identify a critical limitation of this inherent reward design: both $\log(\pi_T/\pi_\theta)$ and $\log(\pi_\theta/\pi_T)$ are unbounded and can be either positive or negative. However, a key empirical principle shared by mainstream RLVR algorithms (Schulman et al., 2017; Guo et al., 2025; Shao et al., 2024) is that all tokens leading to a correct final outcome should receive non-negative advantages, while tokens leading to an incorrect outcome should receive non-positive advantages. This principle ensures that every token in a correct trajectory is treated as a valid prediction and reinforced accordingly, while every token in an incorrect trajectory is treated as an erroneous prediction and suppressed accordingly. Sampled-token OPD violates this principle: on correct trajectories, a negative value of $\log(\pi_T/\pi_\theta)$ penalizes valid token behaviors, while on incorrect trajectories, a negative value of $\log(\pi_\theta/\pi_T)$ encourages erroneous token predictions.

Motivated by this gap, we use a ReLU gating mechanism to standardize the reward signs according to trajectory correctness: it enforces non-negative $\log(\pi_T/\pi_\theta)$ values for all verified correct trajectories and ensures non-negative $\log(\pi_\theta/\pi_T)$ values for all incorrect trajectories. This minimal correction directly converts conventional sampled-token OPD into a valid RLVR method while preserving the guidance from the teacher model. Specifically, for correct trajectories, larger $\log(\pi_T/\pi_\theta)$ values—indicating accurate student predictions where the teacher is more confident than the student—yield stronger rewards. For incorrect trajectories, larger $\log(\pi_\theta/\pi_T)$ values—corresponding to wrong student predictions where the student is more confident than the teacher—incur heavier penalties. This learning paradigm aligns well with human learning intuition, which **reinforces reliable correct behaviors and suppresses overconfident erroneous predictions**. Furthermore, by reformulating sampled-token OPD into an RLVR formulation, our OPDVR supports seamless integration with existing prominent RL algorithms, such as GRPO, DAPO and PPO. To instantiate this, we combine OPDVR with GRPO to obtain a new variant, which we call **Group Relative Policy Distillation (GRPD)**. Overall, our method endows the model with dual supervisory signals: the rigorous task-level verifiable correctness from RLVR and the dense token-level distributional guidance from teacher distillation, leading to more robust and effective post-training optimization for reasoning tasks. Our experiments show that both OPDVR and GRPD consistently outperform OPD across all benchmarks.

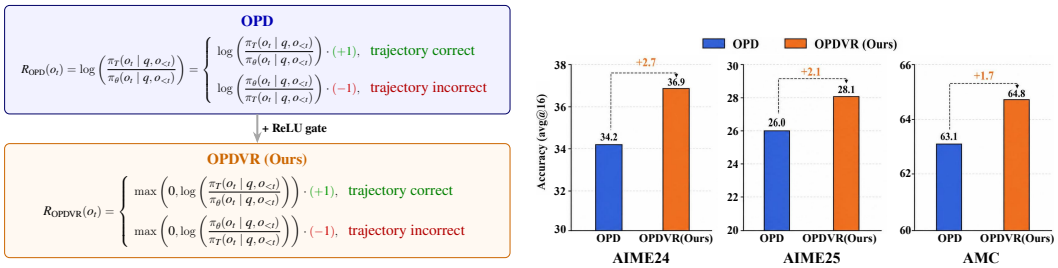


Figure 1: Left: Overview of OPDVR. The ReLU gating mechanism ensures correct trajectories receive non-negative rewards and incorrect ones receive non-positive rewards, while preserving the teacher's distributional guidance. Right: Results on AIME24, AIME25, and AMC under the same-architecture setting (Qwen3-4B ← Qwen3-4B-RL), reported as avg@16 accuracy.

2 Related Work

Reinforcement Learning with Verifiable Rewards and On-policy Distillation. Reinforcement Learning with Verifiable Rewards (RLVR) has become a highly effective post-training paradigm for reasoning tasks, using rule-based verifiers to provide objective outcome signals such as answer correctness or code compilation (Schulman et al., 2017; Jaech et al., 2024; Guo et al., 2025; Shao et al., 2024). However, these rewards are sparse, creating a severe credit assignment problem. On-policy Distillation (OPD) addresses this by providing dense, token-level supervision from a teacher model (Xu et al., 2026a; Xiao et al., 2026; Yang et al., 2025a; Zeng et al., 2026). Yet OPD is purely distributional: it drives the student to mimic the teacher without considering whether the generated response is actually correct, limiting its performance to that of the teacher. Combining OPD's dense

guidance with RLVR's task-level verifiability is thus a natural and promising direction. More related works on RLVR and OPD are discussed in Appendix A.

Combining OPD with RLVR Given the complementary strengths of OPD (dense token-level guidance) and RLVR (task-level correctness), several recent works have attempted to combine them. These include directly weighting the two objectives (Hubotter et al., 2026), applying OPD and RLVR separately depending on the sign of the advantage (Wang et al., 2026; Cai et al., 2026), aligning outcome rewards with process-level distillation signals (Hou et al., 2026), and leveraging teacher-student probability ratios to weight the advantage in GRPO (Yang et al., 2026a). Despite their empirical gains, these approaches often treat OPD and RLVR as distinct objectives or losses that must be either explicitly combined or selectively applied based on heuristic criteria, often introducing additional hyperparameters for balancing the two terms, or rely on heuristic trade-offs. Instead, our method implicitly transforms sampled-token OPD into a proper RLVR method using a simple gated mechanism, while preserving the teacher's distributional guidance—achieving the benefits of both paradigms without explicit multi-objective balancing.

Identifying the Misalignment between Teacher and Verifier Several recent works have identified the inconsistency between teacher distillation signals and verifier feedback. Uni-OPD (Hou et al., 2026) discards prompts whose teacher sequence-level reward gap between positive and negative trajectories is smaller than a safety margin. RG-OPD (Akhondzadeh et al., 2026) directly discards entire trajectories where the teacher signal disagrees with the verifier. However, they both operate at the trajectory level, discarding potentially valuable tokens in inconsistent trajectories. SG-OPD (Xu et al., 2026b) moves to the token level by extrapolating the distillation signal on agreeing tokens and interpolating on conflicting ones. Yet, it only applies a scalar coefficient to scale the conflicting signals, without actually eliminating their undesirable effects. In contrast, we show that a simple ReLU gate on the sampled token suffices to remove conflicting signals, directly transforming sampled-token OPD into a proper RLVR method that reinforces reliable correct predictions and suppresses overconfident errors, without introducing any extra hyperparameters or heuristics.

3 PRELIMINARIES

3.1 REINFORCEMENT LEARNING WITH VERIFIABLE REWARDS (RLVR)

Reinforcement Learning with Verifiable Rewards (RLVR) optimizes a policy using automatically verifiable reward signals, such as whether a mathematical answer is correct. For a policy π_θ generating a response o given prompt q, the objective is:

$$\mathcal{J}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[R\right],$$

where R is the trajectory-level reward ($R = +1/-1$ or $1/0$). Using the REINFORCE trick, its gradient is:

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[R \cdot \sum_{t=1}^{|o|} \nabla_\theta \log \pi_\theta(o_t \mid q, o_{<t})\right].$$

This yields the RLVR loss:

$$\mathcal{L}_{\text{RLVR}}(\theta) = -R \cdot \sum_{t=1}^{|o|} \log \pi_\theta(o_t \mid q, o_{<t}).$$

REINFORCE suffers from high variance. Group Relative Policy Optimization (GRPO) (Guo et al., 2025; Shao et al., 2024) reduces variance by normalizing rewards within a group of G responses:

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

The on-policy GRPO loss (without clipping or KL penalty) is:

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\hat{A}_{i,t}\log \pi_\theta(o_{i,t} \mid q, o_{i,<t}).$$

3.2 On-policy Distillation

On-policy Distillation (OPD) optimizes a student policy π_θ toward a teacher policy π_T by minimizing their reverse KL divergence:

$$\min_\theta \ \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[\sum_{t=1}^{|o|} D_{\mathrm{KL}}(\pi_\theta(\cdot \mid q, o_{<t}) \,\|\, \pi_T(\cdot \mid q, o_{<t}))\right].$$

The **full-vocabulary OPD** computes this exact KL at every token position:

$$\mathcal{L}_{\mathrm{OPD}}^{\mathrm{full}}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[\sum_{t=1}^{|o|}\sum_{v\in\mathcal{V}} \pi_\theta(v \mid q, o_{<t}) \log \frac{\pi_\theta(v \mid q, o_{<t})}{\pi_T(v \mid q, o_{<t})}\right].$$

This loss is directly minimized. In practice, the log-ratio $\log(\pi_\theta/\pi_T)$ is treated as a constant reward with stop-gradient when computing gradients, which is equivalent to the exact gradient of the reverse KL via the log-derivative trick.

Computing the reverse KL over the full vocabulary at every token position is computationally expensive. A practical compromise is **Top-k OPD**, which restricts the KL computation to the k tokens with the highest student probabilities:

$$\mathcal{L}_{\mathrm{OPD}}^{\text{top-}k}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[\sum_{t=1}^{|o|} D_{\mathrm{KL}}\left(\bar{\pi}_\theta^{(S_t)} \,\|\, \bar{\pi}_T^{(S_t)}\right)\right],$$

where S_t is the set of top-k tokens.

The most widely adopted variant is **sampled-token OPD**. It uses a single Monte Carlo sample $o_t \sim \pi_\theta(\cdot \mid q, o_{<t})$ to approximate the reverse KL gradient:

$$\mathcal{L}_{\mathrm{OPD}}^{\mathrm{sample}}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[\sum_{t=1}^{|o|} \log \frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})}\right].$$

The log-ratio $\log(\pi_\theta/\pi_T)$ is treated as a constant reward with stop-gradient.

4 Method

4.1 Revisiting Sampled-Token OPD from an RLVR Perspective

We begin with the sampled-token OPD loss, which approximates the reverse KL gradient with a single Monte Carlo sample:

$$\mathcal{L}_{\mathrm{OPD}}^{\mathrm{sample}}(\theta) = \mathbb{E}_{o\sim\pi_\theta(\cdot|q)}\left[\sum_{t=1}^{|o|} \log \frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})}\right].$$

Its gradient is:

$$\nabla_\theta \mathcal{L}_{\mathrm{OPD}}^{\mathrm{sample}} = \sum_{t=1}^{|o|} \log \frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})} \cdot \nabla_\theta \log \pi_\theta(o_t \mid q, o_{<t}).$$

Recall the RLVR gradient $-R\sum_t \nabla_\theta \log \pi_\theta(o_t \mid q, o_{<t})$. Matching coefficients gives the implicit token-level reward:

$$R_{\mathrm{OPD}}(o_t) = \log \frac{\pi_T(o_t \mid q, o_{<t})}{\pi_\theta(o_t \mid q, o_{<t})} = -\log \frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})}.$$

However, unlike RLVR, where the reward sign is determined by the verifier outcome, the sign of Ropd is solely determined by the teacher-student probability ratio. To analyze their alignment, we

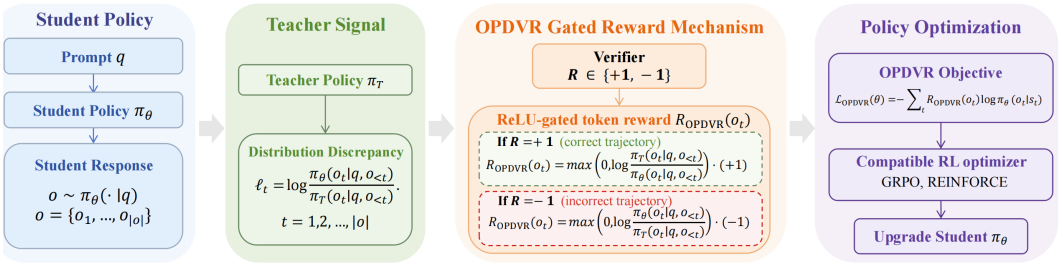


Figure 2: Training pipeline of OPDVR.

consider two cases based on trajectory correctness:

$$R_{\text{OPD}}(o_t) = \begin{cases} \log \dfrac{\pi_T(o_t \mid q, o_{<t})}{\pi_\theta(o_t \mid q, o_{<t})} \cdot (+1), & \text{trajectory correct,} \\ \log \dfrac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})} \cdot (-1), & \text{trajectory incorrect.} \end{cases}$$

In other words, on correct trajectories, the gradient is weighted by $\log(\pi_T/\pi_\theta)$; on incorrect trajectories, it is weighted by $\log(\pi_\theta/\pi_T)$ with a negative sign.

The empirical RLVR principle requires that correct trajectories receive positive rewards and incorrect trajectories receive negative rewards. However, the sign of $\log(\pi_T/\pi_\theta)$ is determined by whether $\pi_T > \pi_\theta$ or $\pi_T < \pi_\theta$, which is independent of trajectory correctness. This creates two failure cases: correct trajectories can receive negative rewards when $\pi_T < \pi_\theta$, and incorrect trajectories can receive positive rewards when $\pi_T > \pi_\theta$. Both violate the RL principle.

4.2 OPDVR: A Simple Gated Mechanism

We propose On-policy Distillation with Verifiable Reward (OPDVR) to resolve this issue. The solution is **extremely simple**: apply a ReLU gate to enforce RLVR compliance on the sampled token while preserving the teacher's distributional guidance.

The reward can be written as:

$$R_{\text{OPDVR}}(o_t) = \begin{cases} \max\left(0, \log \dfrac{\pi_T(o_t \mid q, o_{<t})}{\pi_\theta(o_t \mid q, o_{<t})}\right) \cdot (+1), & \text{trajectory correct,} \\ \max\left(0, \log \dfrac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_T(o_t \mid q, o_{<t})}\right) \cdot (-1), & \text{trajectory incorrect.} \end{cases}$$

The corresponding loss is:

$$\mathcal{L}_{\text{OPDVR}}(\theta) = -\sum_{t=1}^{|o|} R_{\text{OPDVR}}(o_t) \cdot \log \pi_\theta(o_t \mid q, o_{<t}).$$

This gated mechanism ensures that:

- Correct trajectories receive non-negative rewards, and incorrect trajectories receive non-positive rewards, aligning the reward sign with trajectory correctness.
- For tokens on correct trajectories, the reward magnitude is larger when the teacher is more confident than the student, i.e., $\log(\pi_T/\pi_\theta)$ is larger. This encourages the model to reinforce choices that are both correct and reliable according to the teacher.
- For tokens on incorrect trajectories, the penalty magnitude is larger when the student is more confident than the teacher, i.e., $\log(\pi_\theta/\pi_T)$ is larger. This forces the model to suppress overconfident mistakes.

4.3 Interpreting the ReLU Gating Mechanism as a Conditional Mask

In this subsection, we interpret the ReLU gating mechanism in OPDVR as a *conditional token mask*: it selectively zeroes out gradient updates on tokens whose learning direction conflicts with the verifier signal. Standard sampled-token OPD lacks this mask, so it inevitably updates all tokens based solely on the teacher-student confidence ratio, regardless of trajectory correctness.

To see this, recall the standard OPD gradient for a sampled token a:

$$g_{\mathrm{OPD}} = \underbrace{\left[\log \frac{\pi_T}{\pi_\theta}\right]_+ \nabla_\theta \log \pi_\theta(a)}_{\text{Term A: push up (teacher more confident)}} - \underbrace{\left[\log \frac{\pi_\theta}{\pi_T}\right]_+ \nabla_\theta \log \pi_\theta(a)}_{\text{Term B: pull down (student more confident)}}. \tag{1}$$

The OPDVR gradient instead applies the ReLU gate conditioned on the verifier reward $R \in \{+1, -1\}$:

$$g_{\mathrm{OPDVR}} = \begin{cases} \text{Term A}, & R = +1, \\ -\text{Term B}, & R = -1. \end{cases} \tag{2}$$

Thus, compared to standard OPD, OPDVR *removes* exactly two types of tokens whose updates conflict with the verifier:

- **Type I Conflicting Token (Correct Trajectory, $\pi_\theta > \pi_T$):** When the trajectory is correct ($R = +1$), the student has already produced the right answer. If the student is more confident than the teacher on a specific token ($\pi_\theta > \pi_T$), then Term B is activated. Standard OPD applies a *negative* gradient, pulling down this correct token's probability to match the teacher's lower confidence. This update works against the verifier signal in two ways:
  1. The student's higher confidence is *consistent* with the correct outcome.
  2. Reducing the student's confidence on a correct answer introduces a distributional distortion that is not driven by task performance.
- **Type II Conflicting Token (Incorrect Trajectory, $\pi_T > \pi_\theta$):** When the trajectory is incorrect ($R = -1$), the student has produced a wrong answer. If the teacher is more confident than the student on a specific token ($\pi_T > \pi_\theta$), then Term A is activated. Standard OPD applies a *positive* gradient, pushing up this incorrect token's probability to align with the teacher. This update works against the verifier signal in two ways:
  1. The teacher's high confidence does not accompany a correct trajectory here.
  2. Increasing the probability of tokens on an incorrect trajectory reinforces a reasoning pattern that the verifier has marked as wrong.

By masking out these two token types, OPDVR preserves the student's correct high-confidence predictions and withholds the teacher's guidance on wrong answers. The teacher still controls the *magnitude* of the update via the log-ratio, but the verifier now determines the *direction*—reinforce or suppress—so that the distillation process never works against the task reward. **We provide a theoretical analysis of OPDVR in Appendix B.** Under a soft-value teacher assumption, we show that a single verifier-gated mirror update jointly improves task success and preserves teacher matching: task success does not decrease for any step size, and teacher KL does not increase for a modest step size. This provides a principled justification for why OPDVR can improve task performance without sacrificing distributional alignment.

4.4 Group Relative Policy Distillation (GRPD)

Having established that OPDVR transforms sampled-token OPD into a proper RLVR method with a binary verifier reward, a natural extension is to replace this coarse binary signal with a more nuanced, group-relative advantage estimate like GRPO (Guo et al., 2025; Shao et al., 2024).

Recall from Section 3 that GRPO computes a group-relative advantage $\hat{A}_{i,t}$ for each token position t in response o_i:

$$\hat{A}_{i,t} = \frac{R_i - \mathrm{mean}(\{R_j\}_{j=1}^G)}{\mathrm{std}(\{R_j\}_{j=1}^G)},$$

Table 1: Results on same-architecture distillation. All models are evaluated with avg@16.

Method	AIME24	AIME25	AMC	MATH500	Minerva	OlympiadBench	Avg.
Student (Qwen3-4B)	24.0	15.8	60.8	80.9	27.6	42.9	42.0
Teacher (Qwen3-4B-RL)	36.0	29.0	65.9	87.0	35.4	49.3	50.4
Sampled-Token OPD	34.2	26.0	63.1	85.5	31.6	46.5	47.8
Top-64 OPD	34.6	23.5	62.0	85.0	32.2	46.8	47.4
OPD+GRPO	33.1	24.2	64.4	85.2	32.9	46.5	47.7
RLSD	28.7	20.0	62.1	82.8	31.2	44.0	44.8
ExOPD	29.8	26.9	62.2	**86.6**	**35.9**	46.6	47.9
SG-OPD	31.6	26.2	61.2	84.7	35.2	45.8	47.5
OPDVR (Ours)	**36.9**	**28.1**	**64.8**	84.7	33.2	**47.0**	**49.1**

where $R_i \in \{0, 1\}$ is the verifier reward for response o_i, and G is the group size.

We now apply the same ReLU gating logic, but with the binary correctness sign R replaced by the group-relative advantage $\hat{A}_{i,t}$. The GRPD reward becomes:

$$R_{\text{GRPD}}(o_{i,t}) = \begin{cases} \max\left(0, \log \dfrac{\pi_T(o_{i,t} \mid q, o_{i,<t})}{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}\right) \cdot (+1), & \hat{A}_{i,t} > 0, \\ \max\left(0, \log \dfrac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_T(o_{i,t} \mid q, o_{i,<t})}\right) \cdot (-1), & \hat{A}_{i,t} < 0. \end{cases}$$

Equivalently, this can be written compactly as:

$$R_{\text{GRPD}}(o_{i,t}) = \text{sign}(\hat{A}_{i,t}) \cdot \text{ReLU}\left(\text{sign}(\hat{A}_{i,t}) \cdot \log \frac{\pi_T(o_{i,t} \mid q, o_{i,<t})}{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}\right),$$

where $\text{sign}(\hat{A}_{i,t})$ ensures the direction of the update is governed by the advantage sign.

5 Experiments

5.1 Main Experimental Settings

We conduct experiments on both same-architecture and cross-architecture distillation settings to evaluate the effectiveness of our method.

Same-architecture setting. We use Qwen3-4B-nonthinking as the student model and distill it from a teacher model of the same architecture, which is obtained by training Qwen3-4B with GRPO on the DeepMath dataset (He et al., 2026). Following Yang et al. (2026b), we use the filtered subset of the DeepMath dataset consisting of 57k samples with difficulty level ≥ 6.

Cross-architecture setting. We use the DAPO-Math-17k dataset (Yu et al., 2026). The teacher model is Qwen3-4B-base, fine-tuned with GRPO for 3 epochs on the same dataset. The student model is Qwen3-1.7B-base. The distillation training runs for 3 epochs.

Baselines. We compare against Sampled-Token OPD, Top-64 OPD, OPD+GRPO (Xiao et al., 2026), RLSD (Yang et al., 2026a), ExOPD (Yang et al., 2026b), and SG-OPD (Xu et al., 2026b). Implementation details of all baselines are provided in Appendix F.2.

Benchmarks. We evaluate all models on six reasoning benchmarks: AIME24, AIME25, AMC, MATH500, Minerva, and OlympiadBench.

5.2 Main Results

Tables 1 and 2 report the main results on six reasoning benchmarks. As shown in both tables, OPDVR consistently outperforms all baselines in terms of average accuracy across both distillation settings. In the same-architecture setting, OPDVR achieves the highest average score of 49.1, outperforming the next best method (ExOPD, 47.9) by 1.2 points, and even surpasses the teacher model on AIME24

Table 2: Results on cross-architecture distillation. All models are evaluated with avg@16.

Method	AIME24	AIME25	AMC	MATH500	Minerva	OlympiadBench	Avg.
Student (Qwen3-1.7B-Base)	4.1	1.7	23.2	48.9	8.9	17.1	17.3
Teacher (Qwen3-4B-Base-RL)	10.6	13.1	40.3	74.2	17.2	30.0	30.9
Sampled-Token OPD	6.5	2.1	24.8	59.1	11.5	21.6	20.9
Top-64 OPD	**8.5**	**3.3**	26.4	60.1	10.7	21.4	21.7
OPD+GRPO	8.3	1.7	23.2	58.0	10.2	21.8	20.5
RLSD	6.6	2.9	28.7	59.3	10.4	21.6	21.6
ExOPD	7.0	**3.3**	25.9	60.4	**12.3**	21.4	21.6
SG-OPD	5.5	2.8	28.2	59.7	10.7	22.3	21.5
OPDVR (Ours)	**8.5**	**3.3**	**30.3**	**60.8**	11.6	**22.0**	**22.8**

Table 3: Results on Group Relative Policy Distillation. All models are evaluated with avg@16.

Method	AIME24	AIME25	AMC	MATH500	Minerva	OlympiadBench	Avg.
Student (Qwen3-4B)	24.0	15.8	60.8	80.9	27.6	42.9	42.0
Teacher (Qwen3-4B-RL)	36.0	29.0	65.9	87.0	35.4	49.3	50.4
GRPO	28.3	20.8	62.3	83.9	28.9	44.6	44.8
OPD	32.0	**31.7**	65.6	85.4	28.9	46.6	48.4
Distilled RL	33.1	28.9	66.6	**86.3**	30.3	46.8	48.7
GRPD (Ours)	**34.8**	**31.7**	**67.0**	85.6	**30.5**	**47.0**	**49.4**

(36.9 vs. 36.0). In the cross-architecture setting, OPDVR again achieves the best average score of 22.8, outperforming the strongest baseline (Top-64 OPD, 21.7) by 1.1 points, with notable gains on AMC (30.3 vs. 28.7 for RLSD) and OlympiadBench (22.0 vs. 21.8 for OPD+GRPO), highlighting its robustness to both architectural differences and diverse baseline designs.

5.3 Group Relative Policy Distillation

To further validate the effectiveness of replacing the binary verifier reward with group-relative advantages, we instantiate OPDVR with GRPO-style advantage estimation (described in Section 4.4) and evaluate it under the same-architecture setting. To more cleanly evaluate the effectiveness of replacing binary rewards with group-relative advantages, we conduct this experiment on DAPO-Math-17K, distinct from the teacher's DeepMath training data. The group size is set to $G = 8$. We compare GRPD against GRPO, standard sampled-token OPD and Distilled RL (Wang et al., 2026), a method that selectively applies OPD on positive advantages and GRPO on negative advantages. Table 3 reports the results. GRPD outperforms all baselines across all six benchmarks.

5.4 Ablation Study: Inverse-Gated Experiment

To examine the effectiveness of the gating mechanism in OPDVR, we conduct an inverse-gated ablation under the same-architecture setting (Qwen3-4B $\leftarrow$ Qwen3-4B-RL) on DeepMath, following the same experimental setup as the main experiments. Recall that OPDVR keeps a token only when its update direction agrees with the verifier signal: on a correct trajectory ($R{=}+1$), tokens where the teacher is more confident than the student ($\pi_T > \pi_\theta$) are kept and tokens where the student is already more confident ($\pi_\theta > \pi_T$) are gated out; on an incorrect trajectory ($R{=}-1$), the roles are reversed – tokens where the student is more confident than the teacher are kept, and tokens where the teacher is more confident are gated out. The inverse-gated variant swaps these two sets exactly: it keeps the tokens that OPDVR gates out (the student-more-confident tokens on correct trajectories and the teacher-more-confident tokens on incorrect trajectories), and gates out the tokens that OPDVR keeps, while keeping the masking ratio and everything else unchanged.

Table 4 shows the final benchmark results, and Figure 3 tracks the accuracy reward on the training set. The three variants start from the same point and separate monotonically throughout training, yielding a consistent ordering OPDVR $>$ OPD $>$ Inverse-Gated in both the training curves and the final benchmarks. Reversing the gate falls below vanilla OPD on all six benchmarks, confirming the effectiveness of the gating mechanism. Notably, even the inverse-gated variant still improves over the initial model, indicating that the teacher's distributional guidance remains useful—though its benefits

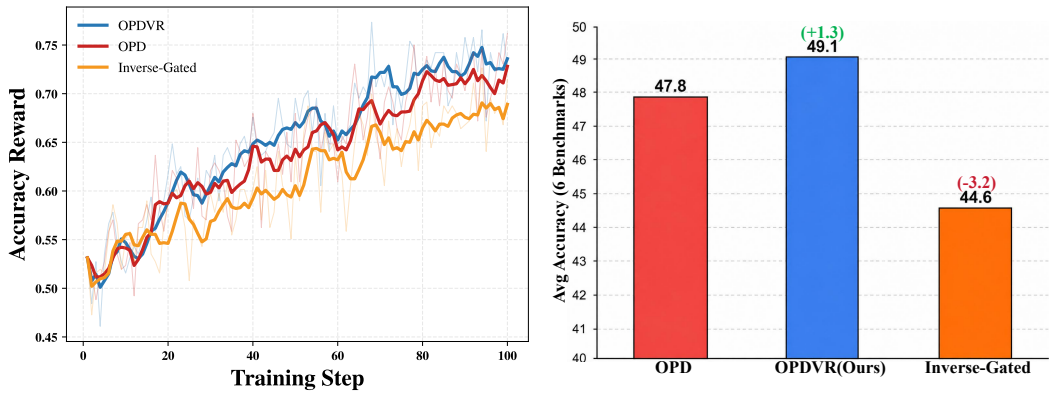


Figure 3: Ablation study on the gating mechanism. **Left**: training-time accuracy reward of OPD, OPDVR, and the inverse-gated variant. **Right**: average accuracy over six benchmarks.

Table 4: Ablation study on the gating mechanism. We compare OPD, OPDVR, and an inverse-gated variant where the ReLU gate is applied in the opposite direction.

Method	AIME24	AIME25	AMC	MATH500	Minerva	OlympiadBench	Avg.
Student (Qwen3-4B)	24.0	15.8	60.8	80.9	27.6	42.9	42.0
Teacher (Qwen3-4B-RL)	36.0	29.0	65.9	87.0	35.4	49.3	50.4
OPD	34.2	26.0	63.1	85.5	31.6	46.5	47.8
OPDVR (Ours)	36.9	28.1	64.8	84.7	33.2	47.0	49.1
Inverse-Gated	30.3	21.2	62.3	83.6	27.7	42.8	44.6

are substantially hindered when the sign is misaligned with trajectory correctness. More training dynamics are provided in Appendix E.

6 Conclusion

We revisited sampled-token OPD from an RLVR perspective and observed that its implicit reward is governed by the teacher-student probability ratio rather than trajectory correctness, which can lead to updates that can penalize tokens in correct trajectories or reward tokens in incorrect ones. To address this, we proposed OPDVR, which adds a simple ReLU gate on the sampled token to enforce RLVR compliance. This minimal modification combines OPD and RLVR without any hyperparameter or heuristic trade-off. Theoretically, we show that under a soft-value teacher assumption, a single verifier-gated mirror update jointly improves task success and preserves teacher matching. Experiments across mathematical reasoning benchmarks demonstrate that OPDVR consistently outperforms standard OPD.

More importantly, by reformulating OPD as an RLVR method, OPDVR becomes a general framework that can be readily integrated with any policy gradient algorithm, including REINFORCE, GRPO, and DAPO. We instantiate this with GRPO and present Group Relative Policy Distillation (GRPD), which further demonstrates the versatility and strength of our approach. The empirical success of both OPDVR and GRPD across multiple benchmarks confirms that aligning distillation signals with trajectory correctness is not only effective but also broadly applicable, offering a simple yet powerful recipe for future post-training methods.

AI USE STATEMENT

In this work, we used generative AI tools for the following required-disclosure tasks: implement methods, clean and reformat dataset, assist in the writing of proofs, provide critical ingredients for proving mathematical claims, interpret results, assist with translation, support qualitative and thematic data analysis and formulate mathematical claims.

We have not used generative AI tools for the following other required-disclosure tasks that were relevant to our research but performed entirely by the authors: help develop theoretical models or conceptual frameworks, propose or refine hypotheses, design or provide feedback on research methodology or experiments.

The remaining required-disclosure tasks are not applicable to this work because our study did not involve them: generate synthetic data sets.

Additionally, we used generative AI tools for the following recommended-disclosure tasks: draft parts of a research paper, edit a research paper to improve readability, format references, and create or edit software code (for utility scripts and visualisation functions).

We have reviewed all AI-assisted work. Specifically:

- All AI-generated code was tested on multiple runs, verified for correctness against expected outputs, and peer-reviewed by at least two authors.
- The cleaned dataset was manually inspected for consistency and missing-value handling after AI-assisted preprocessing.
- AI-suggested steps in mathematical proofs were independently re-derived and cross-checked by the authors.
- AI-drafted text was substantially revised to match our scientific voice and checked for factual accuracy against our experimental results.

We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI. The core research ideas, experimental design, and final interpretation of results were performed solely by the authors without AI involvement.

ETHICS STATEMENT

This paper studies post-training algorithms for LLMs. It does not involve human subjects, personal data, or privacy-sensitive information. All datasets used are publicly available and contain mathematical problems. We foresee no ethical concerns and declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

We describe our method in detail in Section 4. Full implementation details, including hyperparameters and training configurations are provided in Appendix F.

REFERENCES

Mohammad Sadegh Akhondzadeh, Vijay Lingam, Atula Tejaswi, Chanakya Ekbote, Sujay Sanghavi, and Aleksandar Bojchevski. Reward-gated on-policy distillation. *arXiv preprint arXiv:2607.04037*, 2026.

Qiye Cai, Yichuan Ma, Linyang Li, Peiji Li, Yongkang Chen, Qipeng Guo, Yicheng Zou, Tao Gui, Xiaocheng Feng, and Bing Qin. H2 sd: Hybrid hindsight self-distillation. *arXiv preprint arXiv:2607.18955*, 2026.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2160–2169, 2019. URL https://proceedings.mlr.press/v97/geist19a.html.

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

Hasan Abed Al Kader Hammoud, Kumail Alhamoud, Abed Hammoud, Elie Bou-Zeid, Marzyeh Ghassemi, and Bernard Ghanem. Train long, think short: Curriculum learning for efficient reasoning. *arXiv preprint arXiv:2508.08940*, 2025.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 138306–138322, 2026.

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning. *arXiv preprint arXiv:2504.01296*, 2025.

Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, et al. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *arXiv preprint arXiv:2605.03677*, 2026.

Chengyu Huang, Zhengxin Zhang, and Claire Cardie. Hapo: Training language models to reason concisely via history-aware policy optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 31122–31130, 2026.

Jonas Hubotter, Frederike Lubeck, Lejs Deen Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. *ArXiv*, abs/2601.20802, 2026. URL https://api.semanticscholar.org/CorpusID:285102353.

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.

Yuxian Jiang, Yafu Li, Guanxu Chen, Dongrui Liu, Yu Cheng, and Jing Shao. Rethinking entropy regularization in large reasoning models. *arXiv preprint arXiv:2509.25133*, 2025.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pp. 267–274, 2002.

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*, 2026a.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026b.

Wei Liu, Ruochen Zhou, Yiyun Deng, Yuzhen Huang, Junteng Liu, Yuntian Deng, Yizhe Zhang, and Junxian He. Learn to reason efficiently with adaptive length-based reward shaping. *arXiv preprint arXiv:2505.15612*, 2025.

Minjae Oh, Sangjun Song, Gyubin Choi, Yunho Choi, and Yohan Jo. Kl for a kl: On-policy distillation with control variate baseline. *arXiv preprint arXiv:2605.07865*, 2026.

Aleksei Petrenko, Ben Lipkin, Kevin Chen, Erik Wijmans, Marco Cusumano-Towner, Raja Giryes, and Philipp Krähenbühl. Entropy-preserving reinforcement learning. *arXiv preprint arXiv:2603.11682*, 2026.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897, 2015. URL https://proceedings.mlr.press/v37/schulman15.html.

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.

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

Zhenpeng Su, Leiyu Pan, Minxuan Lv, Yuntao Li, Wenping Hu, Fuzheng Zhang, Kun Gai, and Guorui Zhou. Ce-gppo: Coordinating entropy via gradient-preserving clipping policy optimization in reinforcement learning. *ArXiv*, abs/2509.20712, 2025. URL https://api.semanticscholar.org/CorpusID:281526201.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, et al. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

Chen Wang, Zhaochun Li, Jionghao Bai, Yining Zhang, Hexuan Deng, Ge Lan, and Yue Wang. Distilled reinforcement learning for llm post-training. *arXiv preprint arXiv:2607.17247*, 2026.

Violet Xiang, Chase Blagden, Rafael Rafailov, Nathan Lile, Sang Truong, Chelsea Finn, and Nick Haber. Just enough thinking: Efficient reasoning with adaptive length penalties reinforcement learning. *arXiv preprint arXiv:2506.05256*, 2025.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation. *arXiv preprint arXiv:2606.01249*, 2026.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026a.

Haoran Xu, Hongyu Wang, Yifei Gao, Jiaze Li, Xiaofeng Zhang, and Xiaosong Yuan. Sg-opd: Sign-gated on-policy distillation via sign-consistency gating and phased teacher sampling. *arXiv preprint arXiv:2606.09304*, 2026b.

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

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.

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*, 2026a.

Kai Yang, Xin Xu, Yangkun Chen, Weijie Liu, Jiafei Lyu, Zichuan Lin, Deheng Ye, and Saiyong Yang. Entropic: Towards stable long-term training of llms via entropy stabilization with proportional-integral control. *arXiv preprint arXiv:2511.15248*, 2025b.

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *arXiv preprint arXiv:2602.12125*, 2026b.

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. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

A Related Work

Reinforcement Learning with Verifiable Rewards (RLVR) Reinforcement Learning with Verifiable Rewards (RLVR) has emerged as a highly effective paradigm for post-training large language models (Schulman et al., 2017; Jaech et al., 2024; Trinh et al., 2024; Yang et al., 2024; Guo et al., 2025; Shao et al., 2024). RLVR leverages rule-based verifiers to provide objective, deterministic signals such as answer correctness or code compilation. Various extensions have been proposed to improve RLVR, such as controlling length (Liu et al., 2025; Sui et al., 2025; Xiang et al., 2025; Hammoud et al., 2025; Hou et al., 2025; Huang et al., 2026) and stabilizing entropy (Petrenko et al., 2026; Yang et al., 2025b; Jiang et al., 2025; Su et al., 2025). However, a fundamental limitation persists: the reward is sparse, leading to a severe credit assignment problem.

On-policy Distillation On-policy Distillation (OPD) has recently emerged as an efficient post-training paradigm that provides dense, token-level supervisory signals by distilling a teacher model's output distribution into the student (Xu et al., 2026a; Xiao et al., 2026; Yang et al., 2025a; Zeng et al., 2026). Unlike RLVR's sparse outcome-based rewards, OPD offers fine-grained guidance at every generation step. A growing line of work has explored various aspects of OPD, including its failure modes (Fu et al., 2026; Li et al., 2026b), how to determine which tokens are worth training (Xing et al., 2026; Jin et al., 2026), and its training stability issues in practice (Oh et al., 2026). However, OPD's objective is purely distributional: it drives the student to mimic the teacher's output distribution without considering whether the generated response is correct or incorrect. This observation naturally raises the question of whether OPD can be combined with RLVR, where trajectory correctness provides the ultimate optimization signal. Combining the dense supervision of OPD with the task-level verifiability of RLVR thus represents a promising and increasingly relevant direction.

B Theoretical Analysis

On-policy distillation supervises student-generated prefixes, and OPDVR adds a verifier gate to this feedback. We establish joint improvement of task success and teacher matching for the population mirror update of OPDVR.

Setup. Consider a prefix process of horizon H with finite prompt and action sets, a fixed prompt distribution, and terminal verifier $Y \in \{0, 1\}$. Each state includes the time index. Fix student p and teacher T, both with full support. Define $Q^p(s, a) = \Pr_p(Y = 1 \mid s, a)$, where all subsequent actions follow p, and $J(p) = \Pr_p(Y = 1)$. With $z(s, a) = \log[T(a \mid s)/p(a \mid s)]$, the gate is $c(Y, z) = Y[z]_+ - (1 - Y)[-z]_+$, where $[x]_+ = \max\{x, 0\}$. Its conditional mean $\bar{c}(s, a) = \mathbb{E}[c(Y, z(s, a)) \mid s, a]$ is

$$\bar{c}(s, a) = Q^p(s, a)[z(s, a)]_+ - \big(1 - Q^p(s, a)\big)[-z(s, a)]_+. \tag{3}$$

Assumption 1 (Soft-value teacher)**.** *At every reachable prefix s, there exists $\beta_s > 0$ such that*

$$T(a \mid s) = \frac{p(a \mid s)e^{\beta_s Q^p(s,a)}}{Z_s}, \qquad Z_s = \mathbb{E}_{a\sim p(\cdot|s)} e^{\beta_s Q^p(s,a)}. \tag{4}$$

This is the form of a KL-regularized policy improvement (Geist et al., 2019). It is a condition on the current pair (p, T) and is not implied by higher aggregate teacher accuracy.

Mirror update. Apply the standard KL mirror step (Geist et al., 2019, §5) simultaneously at all reachable prefixes,

$$p^+(a \mid s) = \frac{p(a \mid s)e^{\eta \bar{c}(s,a)}}{\sum_{a'} p(a' \mid s)e^{\eta \bar{c}(s,a')}}, \tag{5}$$

holding the teacher and the pre-update quantities $Q^p, z, \bar{c}$ fixed. The next two results characterize this update and bound its effect on the two objectives.

Proposition B.1 (KL-regularized characterization)**.** *Fix a reachable prefix s. Under Assumption 1, the teacher $T(\cdot \mid s)$ and the updated policy $p^+(\cdot \mid s)$ in Equation 5 are the unique maximizers over action distributions r,*

$$T(\cdot \mid s) = \arg\max_r \Big[\mathbb{E}_r Q^p(s,\cdot) - \beta_s^{-1} D_{\mathrm{KL}}(r \| p(\cdot \mid s))\Big], \tag{6}$$

$$p^+(\cdot \mid s) = \arg\max_r \Big[\mathbb{E}_r \bar{c}(s,\cdot) - \eta^{-1} D_{\mathrm{KL}}(r \| p(\cdot \mid s))\Big]. \tag{7}$$

Theorem B.2 (Joint improvement)**.** *Under Assumption 1, the mirror update Equation 5 satisfies:*

(i) for every $\eta > 0$, task success does not decrease,

$$J(p^+) \geq J(p); \tag{8}$$

(ii) if in addition $\eta \leq 1/2$, teacher KL does not increase at any reachable prefix s,

$$D_{\mathrm{KL}}\big(p^+(\cdot \mid s) \| T(\cdot \mid s)\big) \leq D_{\mathrm{KL}}\big(p(\cdot \mid s) \| T(\cdot \mid s)\big). \tag{9}$$

If $Q^p(s,\cdot)$ is nonconstant at a prefix visited with positive probability under p, the inequality in (i) is strict, and that in (ii) is strict at that prefix.

A single verifier-gated on-policy step advances task accuracy and teacher matching together, not one at the expense of the other. Accuracy improves at any step size, and teacher matching is preserved for a modest step. This is the behaviour OPDVR targets. Proofs are in Appendix C.

C Proofs

Equation 3 is immediate from $\mathbb{E}[Y \mid s, a] = Q^p(s,a)$. Because $z(s_t, a_t)$ and the score $\psi_t = \nabla_\theta \log p_\theta(a_t \mid s_t)$ of any smooth parameterization ($p = p_{\theta_0}$) are determined by (s_t, a_t), the tower property gives

$$\mathbb{E}_{\tau\sim p}\Big[\sum_t c(Y, z(s_t,a_t))\,\psi_t\Big] = \mathbb{E}_{\tau\sim p}\Big[\sum_t \bar{c}(s_t,a_t)\,\psi_t\Big],$$

so the population signal analysed below is exactly the one the sampled update follows in expectation; Q^p enters only through the analysis.

C.1 Proof of Proposition B.1

Fix a reachable prefix s and use the local notation $p(a) = p(a \mid s)$, $T(a) = T(a \mid s)$, $Q(a) = Q^p(s,a)$, $z(a) = z(s,a)$, $\bar{c}(a) = \bar{c}(s,a)$, $p^+(a) = p^+(a \mid s)$, $\beta = \beta_s$, and $Z = Z_s$; this suppresses only the fixed prefix, and full support with finiteness makes every sum finite and every KL divergence

well-defined. The continuation policy in Q remains the pre-update student. Assumption 1 rearranges to $Q(a) = \beta^{-1}\log[T(a)/p(a)] + \beta^{-1}\log Z$, so that for every action distribution r,

$$\begin{aligned}\mathbb{E}_r Q - \beta^{-1} D_{\mathrm{KL}}(r\|p) &= \sum_a r(a)Q(a) - \beta^{-1}\sum_a r(a)\log\frac{r(a)}{p(a)} \\ &= \beta^{-1}\sum_a r(a)\left(\log\frac{T(a)}{p(a)} + \log Z\right) - \beta^{-1}\sum_a r(a)\log\frac{r(a)}{p(a)} \\ &= \beta^{-1}\log Z + \beta^{-1}\sum_a r(a)\log\frac{T(a)}{r(a)} \\ &= \beta^{-1}\log Z - \beta^{-1} D_{\mathrm{KL}}(r\|T).\end{aligned} \tag{10}$$

Since $D_{\mathrm{KL}}(r\|T) \geq 0$ with equality iff $r = T$, the teacher T uniquely maximizes $\mathbb{E}_r Q - \beta^{-1}D_{\mathrm{KL}}(r\|p)$, which is Equation 6. Next define $W(u) = \sum_a p(a)e^{u\bar{c}(a)}$ for $u \geq 0$. The mirror update rearranges to $\bar{c}(a) = \eta^{-1}\log[p^+(a)/p(a)] + \eta^{-1}\log W(\eta)$, so that the same computation with (T, Z, β) replaced by $(p^+, W(\eta), \eta)$ gives

$$\begin{aligned}\mathbb{E}_r \bar{c} - \eta^{-1} D_{\mathrm{KL}}(r\|p) &= \sum_a r(a)\bar{c}(a) - \eta^{-1}\sum_a r(a)\log\frac{r(a)}{p(a)} \\ &= \eta^{-1}\sum_a r(a)\left(\log\frac{p^+(a)}{p(a)} + \log W(\eta)\right) - \eta^{-1}\sum_a r(a)\log\frac{r(a)}{p(a)} \\ &= \eta^{-1}\log W(\eta) + \eta^{-1}\sum_a r(a)\log\frac{p^+(a)}{r(a)} \\ &= \eta^{-1}\log W(\eta) - \eta^{-1} D_{\mathrm{KL}}(r\|p^+).\end{aligned} \tag{11}$$

Hence p^+ uniquely maximizes $\mathbb{E}_r\bar{c} - \eta^{-1}D_{\mathrm{KL}}(r\|p)$, which is Equation 7; full support makes both maximizers well-defined.

C.2 Proof of Theorem B.2

All quantities $p, T, Q^p, z, \bar{c}$ are fixed at the current iterate. Set $V^p(s) = \mathbb{E}_{a\sim p(\cdot|s)}Q^p(s,a)$. Early termination is padded with absorbing transitions up to horizon H; at terminal states, $V^p(s_{H+1}) = Y(s_{H+1})$. All logarithms are natural, and finiteness and full support justify the finite sums and differentiation used below. We reuse the local prefix notation of Appendix C.1; a monotonicity property and an interpolation path serve both parts (i) and (ii).

Monotonicity of the gated coefficient. With the local prefix notation, set

$$b = \frac{\log \mathbb{E}_p e^{\beta Q}}{\beta}, \qquad z(a) = \beta\big(Q(a) - b\big). \tag{12}$$

Since $Q \in [0,1]$, we have $b \in [0,1]$. Substituting Equation 12 into the gate gives $\bar{c}(a) = g(Q(a))$, where

$$g(q) = \begin{cases}\beta(1-q)(q-b), & q < b, \\ \beta q(q-b), & q \geq b.\end{cases} \tag{13}$$

The function is continuous at b. Away from this point,

$$g'(q) = \begin{cases}\beta(1+b-2q), & q < b, \\ \beta(2q-b), & q > b.\end{cases} \qquad 0 \leq g'(q) \leq 2\beta. \tag{14}$$

Thus g is nondecreasing. When $\bar{c}$ is viewed as a function of $z = \beta(q-b)$, its derivative lies in $[0,2]$. Continuity at the join point extends both properties across it, so $\bar{c}$ is a nondecreasing, 2-Lipschitz function of z on the relevant interval.

For any functions f, h and distribution μ on the action set, with independent $a, a' \sim \mu$, expansion gives the pairwise covariance identity

$$\mathrm{Cov}_\mu(f,h) = \frac{1}{2}\mathbb{E}_{a,a'\sim\mu}\big[(f(a) - f(a'))(h(a) - h(a'))\big]. \tag{15}$$

Monotonicity of g makes $(Q(a)-Q(a'))(\bar{c}(a)-\bar{c}(a'))$ nonnegative. Monotonicity and the Lipschitz bound in z also imply

$$(z(a)-z(a'))(\bar{c}(a)-\bar{c}(a')) \geq \frac{1}{2}\big(\bar{c}(a)-\bar{c}(a')\big)^2. \tag{16}$$

Applying Equation 15 to both products yields, for every μ,

$$\mathrm{Cov}_\mu(Q,\bar{c}) \geq 0, \qquad \mathrm{Cov}_\mu(z,\bar{c}) \geq \frac{1}{2}\,\mathrm{Var}_\mu(\bar{c}). \tag{17}$$

The interpolation path. For the fixed prefix, introduce the interpolation parameter u, distinct from generation time t, and define

$$p_u(a) = \frac{p(a)e^{u\bar{c}(a)}}{W(u)}, \qquad 0 \leq u \leq \eta, \qquad p_0 = p, \quad p_\eta = p^+. \tag{18}$$

The quantities $Q, z, \bar{c}, T$ are held fixed along this path, and p_u has full support for every finite u. Since $\log p_u(a) = \log p(a) + u\bar{c}(a) - \log W(u)$ and $\partial_u \log W(u) = W'(u)/W(u) = \sum_a p_u(a)\bar{c}(a) = \mathbb{E}_{p_u}\bar{c}$,

$$\partial_u \log p_u(a) = \bar{c}(a) - \partial_u \log W(u) = \bar{c}(a) - \mathbb{E}_{p_u}\bar{c}. \tag{19}$$

Using $\partial_u p_u(a) = p_u(a)\,\partial_u \log p_u(a)$, every fixed function f satisfies

$$\begin{aligned}\frac{d}{du}\mathbb{E}_{p_u} f &= \sum_a f(a)\,\partial_u p_u(a) = \sum_a f(a)\,p_u(a)\big(\bar{c}(a) - \mathbb{E}_{p_u}\bar{c}\big) \\ &= \mathbb{E}_{p_u}(f\bar{c}) - \mathbb{E}_{p_u} f\,\mathbb{E}_{p_u}\bar{c} = \mathrm{Cov}_{p_u}(f,\bar{c}).\end{aligned} \tag{20}$$

If Q is constant, Assumption 1 gives $T = p$, $\bar{c} = 0$, and $p^+ = p$, so parts (i) and (ii) hold with equality; otherwise $\min_a Q(a) < b < \max_a Q(a)$ forces $0 < b < 1$, the derivative in Equation 14 is then strictly positive on each interval, so g is strictly increasing, and every p_u has full support, giving $\mathrm{Cov}_{p_u}(Q,\bar{c}) > 0$ and $\mathrm{Var}_{p_u}(\bar{c}) > 0$.

(i) Task success does not decrease. Taking $f = Q$ in Equation 20 and applying Equation 17 give $\frac{d}{du}\mathbb{E}_{p_u} Q = \mathrm{Cov}_{p_u}(Q,\bar{c}) \geq 0$, so $u \mapsto \mathbb{E}_{p_u} Q$ is nondecreasing on $[0,\eta]$; evaluating at the endpoints $u = 0$ and $u = \eta$ yields

$$\mathbb{E}_{p^+(\cdot|s)} Q^p(s,\cdot) \geq \mathbb{E}_{p(\cdot|s)} Q^p(s,\cdot) = V^p(s). \tag{21}$$

Return to the full prefix notation. Let d_t^r be the distribution of s_t under policy r, with the fixed prompt distribution. The terminal convention above and the Bellman relation give $Q^p(s_t,a_t) = \mathbb{E}[V^p(s_{t+1}) \mid s_t, a_t]$, so under $\tau \sim p^+$ each $Q^p(s_t,a_t)$ may be replaced by $V^p(s_{t+1})$ in expectation; telescoping then gives the finite-horizon performance-difference identity (Kakade & Langford, 2002; see also Schulman et al., 2015, §2),

$$\begin{aligned}&\mathbb{E}_{\tau\sim p^+}\sum_{t=1}^{H}\big(Q^p(s_t,a_t) - V^p(s_t)\big) \\ &\quad= \mathbb{E}_{\tau\sim p^+}\sum_{t=1}^{H}\big(V^p(s_{t+1}) - V^p(s_t)\big) \\ &\quad= \mathbb{E}_{\tau\sim p^+}\big(V^p(s_{H+1}) - V^p(s_1)\big) = J(p^+) - J(p).\end{aligned} \tag{22}$$

The last equality uses the terminal convention $\mathbb{E}_{\tau\sim p^+} V^p(s_{H+1}) = \mathbb{E}_{\tau\sim p^+} Y = J(p^+)$, and the fact that s_1 follows the fixed prompt distribution independently of the policy, so $\mathbb{E}_{\tau\sim p^+} V^p(s_1) = \mathbb{E}_{s_1} V^p(s_1) = J(p)$. Conditioning each summand on s_t yields

$$\begin{aligned}&J(p^+) - J(p) \\ &\quad= \sum_{t=1}^{H}\mathbb{E}_{s\sim d_t^{p^+}}\left[\mathbb{E}_{a\sim p^+(\cdot|s)} Q^p(s,a) - V^p(s)\right].\end{aligned} \tag{23}$$

All terms inside the expectation are nonnegative by Equation 21, proving $J(p^+) \geq J(p)$ for every $\eta > 0$; the distribution here is the *new* visitation distribution $d_t^{p^+}$, so the conclusion concerns actual success after the simultaneous update. Moreover p^+ has full support by Equation 5, so every prefix visited with positive probability under p is also visited under p^+; if $Q^p(s,\cdot)$ is nonconstant at such a prefix, Equation 21 is strict there, the corresponding bracket in Equation 23 is strictly positive, and hence $J(p^+) > J(p)$.

(ii) Teacher KL does not increase at any reachable prefix. Let $K_s(u) = D_{\mathrm{KL}}(p_u \| T) = \mathbb{E}_{p_u} \log \frac{p_u}{T}$. Since $T(a) = p(a)e^{z(a)}$, the log-density ratio is

$$\log \frac{p_u(a)}{T(a)} = \log \frac{p(a)e^{u\bar{c}(a)}/W(u)}{p(a)e^{z(a)}} = -z(a) + u\bar{c}(a) - \log W(u). \tag{24}$$

Differentiating K_s splits into a measure term and an integrand term; by Equation 20 the former is a covariance with $\bar{c}$, and by Equation 19 the latter, $\mathbb{E}_{p_u}\left[\partial_u \log \frac{p_u}{T}\right] = \mathbb{E}_{p_u}\left(\bar{c} - \mathbb{E}_{p_u}\bar{c}\right) = 0$, vanishes. Using Equation 24 and dropping the action-independent term $\log W(u)$,

$$\begin{aligned} K_s'(u) &= \mathrm{Cov}_{p_u}\left(\log \frac{p_u}{T}, \bar{c}\right) + \mathbb{E}_{p_u}\left[\partial_u \log \frac{p_u}{T}\right] = \mathrm{Cov}_{p_u}(-z + u\bar{c}, \bar{c}) \\ &= u \,\mathrm{Var}_{p_u}(\bar{c}) - \mathrm{Cov}_{p_u}(z, \bar{c}) \le \left(u - \tfrac{1}{2}\right) \mathrm{Var}_{p_u}(\bar{c}), \end{aligned} \tag{25}$$

the last step by Equation 17. For $0 < \eta \le 1/2$ the derivative is nonpositive throughout $[0, \eta]$, so $K_s(\eta) \le K_s(0)$, that is, $D_{\mathrm{KL}}\big(p^+(\cdot \mid s) \| T(\cdot \mid s)\big) \le D_{\mathrm{KL}}\big(p(\cdot \mid s) \| T(\cdot \mid s)\big)$. If Q is nonconstant, then $\mathrm{Var}_{p_u}(\bar{c}) > 0$, so $K_s'(u) < 0$ for all $0 \le u < \eta$ and the decrease is strict, including when $\eta = 1/2$. This completes the proof of both parts of Theorem B.2. □

D Directional Alignment and Conflict Removal

We now provide a formal characterization of the advantage of OPDVR over the sampled-token OPD objective. Throughout this section, let $s_t = (q, o_{<t})$ denote the state at step t, $a_t = o_t$ the sampled action, and define the token-level teacher-student log-ratio as

$$r_t = \log \frac{\pi_T(a_t \mid s_t)}{\pi_\theta(a_t \mid s_t)}. \tag{26}$$

As before, $R \in \{+1, -1\}$ denotes the trajectory-level verifier reward.

D.1 Directional Alignment with the Verifier Gradient

Let $u_t = \nabla_\theta \log \pi_\theta(a_t \mid s_t)$ denote the policy gradient direction of the sampled token. The token-level update direction of sampled-token OPD is

$$\Delta_{\mathrm{OPD},t} = r_t u_t, \tag{27}$$

whereas the OPDVR update direction is

$$\Delta_{\mathrm{OPDVR},t} = R\,\mathrm{ReLU}(Rr_t)\, u_t, \tag{28}$$

with $\mathrm{ReLU}(x) = \max(0, x)$.

Recall that the pure verifier-driven RLVR update direction is $\Delta_{\mathrm{RLVR},t} = Ru_t$. The following proposition shows that OPDVR is always aligned with this verifier direction, while OPD may be anti-aligned.

Proposition D.1 (Verifier alignment)**.** *For any token with $u_t \neq 0$, the OPDVR update is never in the opposite direction of the verifier gradient. Specifically,*

$$\langle \Delta_{\mathrm{OPDVR},t}, \Delta_{\mathrm{RLVR},t} \rangle \ge 0. \tag{29}$$

In contrast, sampled-token OPD satisfies

$$\langle \Delta_{\mathrm{OPD},t}, \Delta_{\mathrm{RLVR},t} \rangle = r_t R \|u_t\|^2, \tag{30}$$

which becomes negative whenever r_t and R have opposite signs.

Proof. Using Equation 28 and $\Delta_{\mathrm{RLVR},t} = Ru_t$, we have

$$\begin{aligned} \langle \Delta_{\mathrm{OPDVR},t}, \Delta_{\mathrm{RLVR},t} \rangle &= \langle R\,\mathrm{ReLU}(Rr_t)u_t, Ru_t \rangle \\ &= R^2\,\mathrm{ReLU}(Rr_t)\,\|u_t\|^2 \\ &= \mathrm{ReLU}(Rr_t)\,\|u_t\|^2 \ge 0. \end{aligned} \tag{31}$$

For sampled-token OPD, using Equation 27 gives

$$\langle \Delta_{\mathrm{OPD},t}, \Delta_{\mathrm{RLVR},t} \rangle = \langle r_t u_t, Ru_t \rangle = r_t R \|u_t\|^2, \tag{32}$$

which is negative when $r_t R < 0$. □

The condition $r_t R < 0$ corresponds exactly to the two conflict cases identified earlier: (i) a correct trajectory with $\pi_\theta > \pi_T$, and (ii) an incorrect trajectory with $\pi_T > \pi_\theta$. In both cases, OPD pushes the policy in a direction that reduces the verifier reward.

D.2 Decomposition of the OPD Gradient

The gated mechanism can be interpreted as explicitly removing the verifier-conflicting component from the OPD gradient.

Proposition D.2 (Conflict removal)**.** *The sampled-token OPD update can be decomposed as*

$$\Delta_{\mathrm{OPD},t} = \Delta_{\mathrm{OPDVR},t} + \Delta_{\mathrm{conflict},t}, \tag{33}$$

where

$$\Delta_{\mathrm{conflict},t} = r_t \, \mathbf{1}(r_t R < 0) \, u_t. \tag{34}$$

Moreover, the conflict term always has a non-positive projection onto the verifier gradient:

$$\langle \Delta_{\mathrm{conflict},t}, \Delta_{\mathrm{RLVR},t} \rangle = -|r_t| \, \mathbf{1}(r_t R < 0) \, \|u_t\|^2 \leq 0. \tag{35}$$

Proof. When $r_t R \geq 0$, we have $R \, \mathrm{ReLU}(R r_t) = r_t$, so $\Delta_{\mathrm{OPDVR},t} = r_t u_t$ and the conflict term vanishes. When $r_t R < 0$, we have $\mathrm{ReLU}(R r_t) = 0$, so $\Delta_{\mathrm{OPDVR},t} = 0$ and the conflict term equals $\Delta_{\mathrm{OPD},t}$. The projection onto $\Delta_{\mathrm{RLVR},t}$ follows directly. □

Thus OPDVR is not an arbitrary modification of OPD; it is precisely OPD with the harmful verifier-opposing component removed.

D.3 A Simplified Token-Level Analysis: OPDVR Can Strictly Outperform the Teacher

To illustrate the mechanism while retaining the token-level structure of LLM generation, we consider a simplified single-token decision problem. Let s be a fixed prefix. At this prefix, the model must choose between two candidate next tokens: a_1 denotes a correct key token that leads to a correct final answer, and a_2 denotes an incorrect key token that leads to a wrong final answer. The trajectory-level verifier reward is therefore $R(a_1) = +1$ and $R(a_2) = -1$.

Let $\pi_\theta(a_1 \mid s) = q$ and $\pi_T(a_1 \mid s) = p$. Assume the teacher is suboptimal, i.e., $p < \frac{1}{2}$, and that the initial student policy is already better than the teacher, i.e., $q_0 > p$.

Under sampled-token OPD, the expected loss reduces to the reverse KL divergence

$$\mathcal{L}_{\mathrm{OPD}} = D_{\mathrm{KL}}(\pi_\theta(\cdot \mid s) \,\|\, \pi_T(\cdot \mid s)), \tag{36}$$

whose global minimizer is $\pi_\theta(\cdot \mid s) = \pi_T(\cdot \mid s)$. Hence the OPD optimum satisfies $q = p$, with expected verifier reward

$$J_{\mathrm{OPD}} = 2p - 1. \tag{37}$$

Under OPDVR, the expected update for token a_1 is

$$R(a_1) \, \mathrm{ReLU}\left(R(a_1) \log \frac{p}{q_0} \right) = \mathrm{ReLU}\left(\log \frac{p}{q_0} \right) = 0, \tag{38}$$

because $\log(p/q_0) < 0$ since $q_0 > p$. Similarly, for token a_2 we have $R(a_2) = -1$ and

$$\log \frac{1-p}{1-q_0} > 0 \tag{39}$$

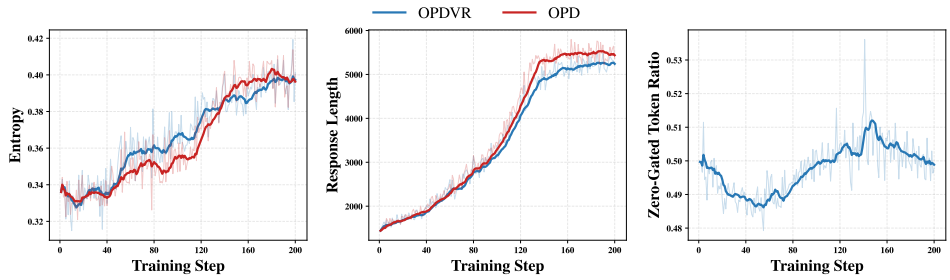
since $q_0 > p$, so the OPDVR update for a_2 is

$$R(a_2) \, \mathrm{ReLU}\left(R(a_2) \log \frac{1-p}{1-q_0} \right) = -\mathrm{ReLU}\left(-\log \frac{1-p}{1-q_0} \right) = 0. \tag{40}$$

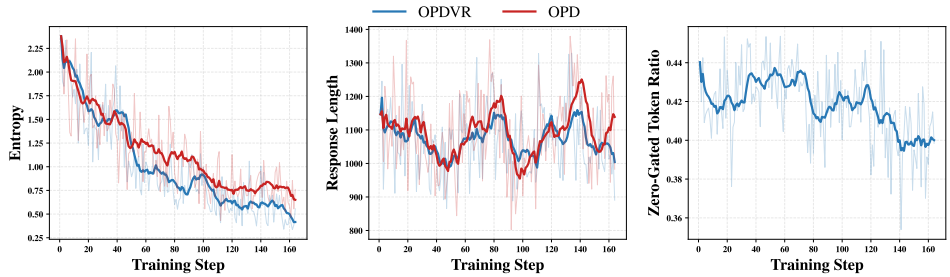
Thus the expected OPDVR gradient vanishes, and the policy remains at q_0. Consequently,

$$J_{\mathrm{OPDVR}} = 2q_0 - 1 > 2p - 1 = J_{\mathrm{OPD}} = J_{\mathrm{teacher}}. \tag{41}$$

Therefore, OPDVR strictly outperforms both standard sampled-token OPD and the teacher policy.



(a) Training dynamic of same-architecture distillation (Qwen3-4B ← Qwen3-4B-RL).



(b) Training dynamic of cross-architecture distillation (Qwen3-1.7B-Base ← Qwen3-4B-Base-RL).

Figure 4: Training dynamics

E Training Dynamics

To understand how token-level gating shapes the distillation process, we track three quantities throughout training: the student policy entropy, the average response length, and the zero-gated token ratio, i.e., the fraction of tokens whose distillation loss is masked by the gate. Figure 4 shows the dynamics of the same-architecture setting (Qwen3-4B ← Qwen3-4B-RL) and the cross-architecture setting (Qwen3-1.7B-Base ← Qwen3-4B-Base-RL), respectively.

The entropy and response-length trajectories of the sampled-token OPD baseline show no consistent pattern across settings; instead, they depend strongly on the teacher–student pair. In the same-architecture setting, the entropy of both methods drifts mildly upward (from $\sim$0.33 to $\sim$0.40) while the response length inflates by more than four times, from roughly 1.6k to over 6.7k tokens. In the cross-architecture setting, the picture inverts: the student entropy collapses rapidly from $\sim$2.0 at the start of training and the response length stays nearly flat throughout. These quantities therefore do not follow a setting-independent trend – their evolution is dictated by the specific teacher and student models rather than by the distillation objective itself.

In contrast, the zero-gated token ratio is consistent across both settings. It remains in a band around fifty percent during the entire course of training ($\approx$0.48–0.50 for the 4B student and $\approx$0.40–0.44 for the 1.7B student), never degenerating toward the trivial extremes of gating all or no tokens. This means that roughly half of the sampled tokens are zeroed by the ReLU gate at any point of training. Recalling the failure cases identified in Section 4, these are exactly the tokens whose implicit OPD reward pushes against the RLVR direction: on correct trajectories where the student already assigns higher probability than the teacher ($\pi_\theta > \pi_T$), and on incorrect trajectories where the teacher still outranks the student ($\pi_T > \pi_\theta$). The stable ratio indicates that such RLVR-violating tokens constitute a persistent fraction of the data throughout training – the gate consistently filters them out while distilling the remaining tokens with their full teacher–student log-ratio magnitude.

F Implementation Details

F.1 Hyperparameters

We provide the detailed hyperparameter configurations used in our experiments in Table 5. All models are trained using the **Verl** (Sheng et al., 2025) framework with the settings specified below.

Table 5: Hyperparameter settings.

Hyperparameter	**Value**
Learning Rate	1e-6
Train Batch Size	256
PPO Mini-Batch Size	256
Max Response Length	8192
Max Prompt Length	1024
Rollout Temperature	1.0
Evaluation Temperature	0.7
Evaluation Top-p	0.95

F.2 Baselines Implementation Details

OPD+GRPO (Xiao et al., 2026). We follow the original OPD objective and add a GRPO-based policy gradient loss with equal weighting (1:1) between the two terms. Specifically, the total loss is $\mathcal{L} = \mathcal{L}_{\text{OPD}} + \mathcal{L}_{\text{GRPO}}$.

RLSD (Yang et al., 2026a). We set the group size to 8. Since we compare against on-policy distillation methods, we replace the original privileged self-model teacher with the same teacher used by OPD to ensure a fair comparison.

ExOPD (Yang et al., 2026b). Following the original paper, we set the extrapolation coefficient to 1.25, which was reported as the best setting.

SG-OPD (Xu et al., 2026b). We set the extrapolation and interpolation coefficients to 1.8 and 1.0, respectively, matching the best configuration reported in the original paper.

G Hardware Setup

All experiments are conducted on NVIDIA GeForce RTX 5090 and NVIDIA H20 GPUs.