Reinforcement Learning from Rich Feedback with Distributional DAgger

Rishabh Agrawal [1] **Jacob Fein-Ashley** [1] **Paria Rashidinejad** [1]

Abstract

Reasoning models have advanced rapidly, but the dominant reinforcement learning from verifiable rewards (RLVR) recipe remains surprisingly narrow: sample many responses and reward each with a single bit indicating whether the final answer is correct. Yet many settings provide *rich feedback*, including execution traces, tool outputs, expert corrections, and model self-evaluations. We study how to use such feedback through a distributional variant of the classic imitation learning algorithm DAgger, where the learner has local access to an expert distribution on states visited by the current policy. This yields a simple forward cross-entropy objective that admits a blackbox expert and whose sequence-level gradient conduct rich credit assignment by propagating future expert-student disagreement back to earlier decisions. We show that prior RL with self-distillation objectives based on reverse KL or Jensen-Shannon fail to guarantee monotonic policy improvement: even when the expert has higher reward, their updates may increase probability on worse actions. In contrast, we show that forward cross-entropy admits monotonic policy improvement and enjoys guarantees on regret. We further show that our objective optimizes a teacher-weighted maximum-likelihood RL lower bound, leading to improved Pass@N. Empirically, our approach, DistIL, improves over RLVR and RL with self-distillation baselines across a variety of domains: scientific reasoning, mathematical reasoning, and coding.

1. Introduction

Reasoning large language models have made significant recent progress (Jaech et al., 2024; Guo et al., 2025; Comanici et al., 2025; Olmo et al., 2025). The dominant paradigm, reinforcement learning from verifiable rewards (RLVR) improves reasoning by sampling many candidate solutions from the model and providing each with a single bit of feedback: whether the final answer is correct. This simple paradigm has been remarkably successful in domains such as coding and mathematical and scientific reasoning tasks where final answers can be checked automatically (Tan et al., 2025; Yue et al., 2025; Zeng et al., 2025; Pan & Liu, 2025; Luo et al., 2025). However, RLVR has important limitations. A scalar terminal reward is typically broadcast across the entire response, causing every token to receive the same weight regardless of its actual contribution. This provides little credit assignment: it does not reveal which tokens, phrases, or reasoning steps caused the success or failure. Moreover, the paradigm's reliance on automatic final-answer verification makes it difficult to apply in unverifiable domains, where final answer correctness cannot be easily checked.

Many important settings, however, provide feedback that is far richer than a single correctness bit: error logs, unit-test results, natural-language critiques, hints, or ground-truth responses. Inspired by the seminal work on knowledge distillation (Hinton et al., 2015), a growing line of research seeks to exploit these signals through on-policy self-distillation, in which the current model is conditioned on feedback to produce a privileged teacher, whose distribution is then distilled back into the original policy (Hübotter et al., 2026; Zhao et al., 2026). This turns sparse outcome supervision into dense token-level guidance. In particular, viewed through the lens of policy gradients, these objectives induce an implicit form of credit assignment: different tokens receive different weights, e.g., determined by the local discrepancy between the student and feedback-conditioned teacher policies (Hübotter et al., 2026).

1.1. Contributions

This motivates us to take a closer look at on-policy self-distillation. Many classical policy-optimization methods—such as conservative policy iteration (Kakade & Langford, 2002), natural policy gradient (Kakade, 2001), and trust-region policy optimization (Schulman et al., 2015)—are designed around *monotonic policy improvement*: under an idealized update, such as a sufficiently small natural-gradient

[1]University of Southern California. Correspondence to: Rishabh Agrawal <rishabha@usc.edu>.

The 3rd *AI for Math Workshop* at the 43rd International Conference on Machine Learning (ICML), Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

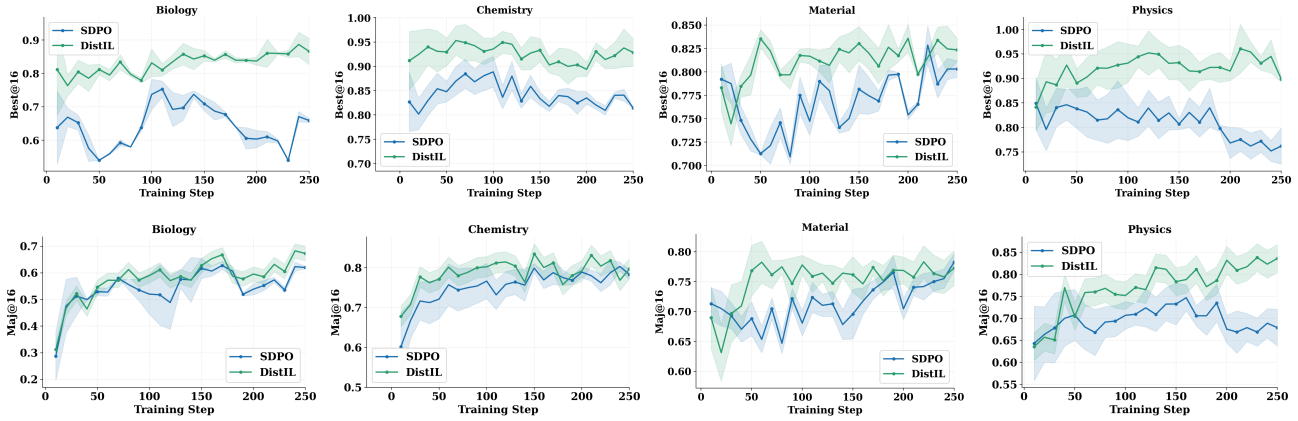

Figure 1. Validation Best@16 (top) and Maj@16 (bottom) over training for RL with self-distillation algorithm SDPO (Hübotter et al., 2026) and DistIL (ours) on Qwen3-8B across four scientific reasoning domains: biology, chemistry, materials, and physics. DistIL generally achieves higher validation performance than SDPO across domains and metrics, with gains often appearing early and sustained during training. SDPO exhibits greater variability with longer training, including a pronounced decline in biology Best@16 after roughly 100 steps and larger oscillations in chemistry and physics; DistIL is comparatively more stable.

or trust-region step, each update should increase policy's average reward. This raises a natural question for on-policy self-distillation. If a feedback-conditioned teacher achieves higher average reward than the student, should an idealized update that pulls the student toward the teacher improve the student? We show that, for on-policy self-distillation objectives with f-divergences, the answer is in general, no. To address this, we revisit the classical DAgger framework (Ross et al., 2011) and propose a distributional variant that enjoys monotonic policy-improvement and regret guarantees, while empirically outperforming existing methods. Concretely, we make the following contributions:

Provable limitations of existing on-policy self-distillation. We begin by analyzing on-policy self-distillation objectives based on general f-divergences, a class that subsumes existing objectives such as reverse-KL divergence (Hübotter et al., 2026) and Jensen–Shannon divergence (Zhao et al., 2026). We identify two limitations. First, we prove that optimizing an f-divergence does not, in general, guarantee monotonic policy improvement; even when the teacher is strictly better than the student and the update is an idealized natural-gradient step, the objective can increase probability on suboptimal actions (Propositions 3.1, 3.2). Second, existing methods typically rely on approximate gradients that provide only *local* credit assignment, measuring teacher–student discrepancy at individual tokens. We show that this local approximation ignores how early token choices shape the future states where teacher–student disagreement appears, erasing precisely the delayed feedback needed to credit earlier decisions. We prove that this can cause learning to converge to strictly suboptimal policies (Proposition 3.3).

DistIL: A new algorithm for RL with rich feedback. To address these limitations, we propose DistIL, a distributional variant of the DAgger imitation-learning algorithm. DistIL optimizes a forward cross-entropy loss between the feedback-conditioned teacher and the student policy, on states visited by the student (c.f. Equation (4)). Unlike reverse-KL or Jensen–Shannon self-distillation objectives, DistIL can leverage teacher log-probabilities when available, but does not require them and its objective can be estimated using samples from the teacher alone, and therefore naturally accommodates black-box teachers such as external models or human experts. Furthermore, DistIL performs *future-aware* credit assignment: teacher–student disagreement at later timesteps can be propagated back to the earlier decisions (c.f. Equation (5)).

DistIL enjoys theoretical guarantees. Beyond algorithmic simplicity, DistIL comes with several theoretical guarantees that explain why the objective is well aligned with reward improvement. First, we show that DistIL's forward cross-entropy loss guarantees monotonic-improvement property missing from prior self-distillation objectives: under standard local assumptions, moving toward a better teacher improves the student's expected reward (Proposition 5.2). Second, we prove that DistIL achieves sublinear regret (Theorem 5.3). Finally, we show that DistIL maximizes a teacher-weighted lower bound on the expected log-likelihood of success, drawing connections to maximum-likelihood RL (Tajwar et al., 2026) and providing a principled explanation for improvements in $\texttt{pass@}N$ for any N (Proposition 5.4, c.f. Figure 1).

Empirical results. The theoretical advantages of DistIL translate into consistent empirical gains. Across scientific reasoning (Feng et al., 2024), coding (Jain et al., 2024), and challenging mathematical reasoning benchmarks, DistIL

outperforms strong RL and self-distillation baselines, including SDPO (Hübotter et al., 2026), OPSD (Zhao et al., 2026), and GRPO (Shao et al., 2024). The gains hold across different feedback regimes, from sparse final-answer correctness to execution traces and ground-truth solutions, highlighting the practical benefit of pairing a reward-aligned objective with future-aware credit assignment.

2. Background and Problem Formulation

Contextual Markov decision process and reinforcement learning. We model consider finite-horizon contextual Markov decision processes as the environment. At the beginning of each episode, a context (i.e., a prompt/problem) $x \sim \rho$ is sampled. The initial state is set to be the context $s_1 = (x, \emptyset)$. At each timestep $t \in [H]$, the policy selects an action $y_t \sim \pi(\cdot \mid s_t)$, and the environment transitions to the next state. In the autoregressive language-model setting considered in this work, the transition is deterministic and the next state is obtained by appending the sampled action to the current prefix. Thus, we write $s_t = (x, y_{1:t-1})$. For simplicity, we sometimes write the entire trajectory as $y = (y_1, \ldots, y_H)$.

Given per-step rewards $r_t(x, y_{1:t})$, the goal of reinforcement learning is to learn a policy maximizing expected cumulative reward:

$$J(\pi) := \mathbb{E}_{x\sim\rho}\mathbb{E}_{y\sim\pi(\cdot|x)}\left[\sum_{t=1}^{H} r_t(x, y_{1:t})\right]. \tag{1}$$

Reinforcement learning with verifiable rewards (RLVR). In many reasoning tasks, rewards can be obtained from automatic verifiers: code can be checked with unit tests, while in some mathematical and scientific problems final answer correctness can be checked. This gives rise to reinforcement learning with verifiable rewards (RLVR), where supervision is typically sparse and outcome-based: $r_t = 0$ for $t < H$, and the terminal reward r_H evaluates the final answer, often as a binary correctness signal. Methods such as GRPO (Shao et al., 2024) and its variants (Liu et al., 2025; Ahmadian et al., 2024) optimize policies using such rewards. Despite its simplicity and empirical success, RLVR provides only delayed supervision. Furthermore, the terminal reward is broadcast across the response, giving every token the same weight regardless of its role in the reasoning process. This makes credit assignment difficult. Moreover, in group-relative or advantage-based updates (Shao et al., 2024), the learning signal can vanish when all sampled responses receive identical outcomes, such as all correct or all incorrect.

Learning from rich feedback via distillation. In many settings, supervision extends beyond final-answer correctness and includes richer signals such as execution traces, error messages, expert annotations, or feedback from other models. These signals provide more informative guidance and can be leveraged to improve learning. Inspired by knowledge distillation, a recent line of work studies variants of *on-policy distillation*, where a student policy π_S is trained to match a teacher policy π_T along trajectories sampled from the student (Agarwal et al., 2024; Yang et al., 2025a; Xiao et al., 2026; Zeng et al., 2026). In on-policy *self*-distillation, the teacher is derived from the same model but conditioned on additional feedback f, i.e., $\pi_T(\cdot \mid x, y_{1:t-1}) := \pi_\theta(\cdot \mid x, f, y_{1:t-1})$ and $\pi_S(\cdot \mid x, y_{1:t-1}) := \pi_\theta(\cdot \mid x, y_{1:t-1})$, where π_T denotes the feedback-conditioned teacher and π_S denotes the student policy. Given trajectories $y \sim \pi_S(\cdot \mid x)$, training minimizes $\mathcal{L}_{\text{self-distill}}$ given as follows

$$\mathbb{E}_{\substack{x\sim\rho \\ y\sim\pi_S(\cdot|x)}}\left[\sum_{t=1}^{|y|} \mathsf{D}\big(\pi_S(\cdot \mid x, y_{1:t-1}) \,\|\, \pi_T(\cdot \mid x, y_{1:t-1})\big)\right], \tag{2}$$

where D is a divergence measure, such as reverse KL in SDPO (Hübotter et al., 2026) or Jensen–Shannon divergence in OPSD (Zhao et al., 2026).

Problem formulation: RL from rich feedback. Given a dataset of prompts/problems $\mathcal{D}$, the goal is to learn a policy π_θ that maximizes the RL objective 1. However, the reward function r is unknown. To make the problem well defined, on-policy self-distillation thus assume teacher that is on average better than the student policy. Formally, let π_T denote the teacher and π_θ the student, and assume a positive reward gap i.e. $\Delta := \mathbb{E}_{x\sim\rho, y\sim\pi_T(\cdot|x)}[r(x,y)] - \mathbb{E}_{x\sim\rho, y\sim\pi_\theta(\cdot|x)}[r(x,y)] > 0$.

Notation. We use $\mathrm{KL}(p\|q)$ for Kullback–Leibler divergence, $\mathrm{JS}(p\|q)$ for Jensen–Shannon divergence, and $\mathrm{H}^{\times}(p,q) := -\mathbb{E}_p[\log q]$ for cross-entropy. The operator $\mathsf{sg}(\cdot)$ denotes stop-gradient.

3. Limitations of Existing On-Policy Self-Distillation Methods

3.1. Does divergence minimization guarantee monotonic policy improvement?

A central objective in policy optimization is *monotonic policy improvement* (MPI), wherein each update increases expected reward. This property is satisfied by many widely used RL methods such as natural policy gradient (NPG) methods, and approximately by practical variants such as PPO (Schulman et al., 2017) and GRPO (Shao et al., 2024). However, existing self-distillation methods, such as SDPO (reverse KL) (Hübotter et al., 2026) and OPSD (Jensen-Shannon) (Zhao et al., 2026), optimize an f-divergence objective instead. We show that such objectives do not, in general, guarantee monotonic improvement, even in the contextual bandit setting.

Proposition 3.1. *Let π_T be a fixed teacher and π_θ a student policy. For the softmax parameterization, the Natural Policy Gradient (NPG) step minimizing an f-divergence $D_f(\pi_T \| \pi_\theta)$, i.e., $\theta' = \theta - \eta\, F(\theta)^{-1} \nabla_\theta D_f(\pi_T \| \pi_\theta)$, satisfies*

$$J(\pi_{\theta'}) = J(\pi_\theta) - \eta\, \mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\left(r(y), g(y)\right) + O(\eta^2),$$

where $g(y) = f\left(\frac{\pi_T(y)}{\pi_\theta(y)}\right) - \frac{\pi_T(y)}{\pi_\theta(y)} f'\left(\frac{\pi_T(y)}{\pi_\theta(y)}\right)$. Consequently, for sufficiently small $\eta > 0$, the NPG step provides a policy improvement if and only if $\mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\left(r(y), g(y)\right) < 0$. In particular, $\Delta > 0$ does not suffice to guarantee this condition.

Appendix B.1 provides a detailed proof of Proposition 3.1.

Remark. Using Proposition 3.1 for SDPO (i.e. reverse KL), we get $\mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\left(r(y), \log \frac{\pi_\theta(y)}{\pi(y)}\right)$ term. The covariance measures whether actions the student overweighs relative to the teacher tend to be high or low reward. If $\mathbb{C}\mathrm{ov} < 0$, excess mass is on bad actions, and the NPG corrects these errors. If $\mathbb{C}\mathrm{ov} > 0$, excess mass is on good actions and the NPG penalizes them, degrading the policy. Notably, the sign of the reward gap Δ does not constrain the sign of this covariance. Since the covariance depends on the joint structure of rewards and the student–teacher density ratio, its sign can vary arbitrarily across policies. As a result, reverse KL (SDPO) updates do not guarantee monotonic improvement.

Proposition 3.1 shows that no guarantees of monotonic policy improvement can be provided. To illustrate that such cases indeed exist, we present Proposition 3.2 below.

Proposition 3.2. *There exists a contextual bandit setting such that, even when the teacher policy is strictly better than the student policy (i.e., $\Delta > 0$), an NPG step minimizing the reverse KL divergence, as in SDPO, does not lead to policy improvement.*

The detailed construction for Proposition 3.2 can be found in Appendix B.2. We construct a simple three-armed bandit where the student over-assigns mass to a suboptimal action; under a reverse-KL NPG update, the normalization term outweighs the local log-ratio, causing the update to further increase probability on this suboptimal action.

3.2. Local credit assignment leads to suboptimal policies

Next, we study the gradient updates used in current self-distillation methods such as SDPO and OPSD. These methods induce *local credit assignment*, whereby gradients are computed token-wise under a frozen trajectory distribution. This arises from the use of a stop-gradient operator in Eq. (2) on the sampling process. Consequently, gradients pass through the expectation and take the following form:

$$\mathbb{E}_{\substack{x\sim\rho \\ y\sim\pi_S(\cdot|x)}} \left[\sum_{t=1}^{|y|} \nabla_\theta D\big(\pi_\theta(\cdot \mid x, y_{1:t-1}) \,\|\, \pi_T(\cdot \mid x, y_{1:t-1})\big)\right], \tag{3}$$

While Eq. (3) simplifies gradient computation, it largely ignores how current token choices affect future token distributions due to the stop-gradient operator, thereby failing to perform proper credit assignment. This mismatch between token-wise and sequence-level gradients can lead to incorrect credit assignment to early decisions, whose impact is only realized downstream. To demonstrate this, we provide a simple construction in Proposition 3.3.

Proposition 3.3. *There exists a two-step MDP and policy π_θ such that the tokenwise gradient provides no update signal for the first-step decision, while the sequence-level gradient is nonzero. Consequently, tokenwise optimization attains a strictly suboptimal solution.*

Proposition 3.3 highlights a fundamental limitation of tokenwise updates: they may fail to assign proper credit to early decisions, leading to strictly suboptimal solutions. Consequently, methods such as SDPO and OPSD, which rely on such gradient estimators, may suffer from this limitation. A concrete construction is provided in Appendix B.3. We construct a simple two-step problem where only one first-step action leads to informative second-step feedback. Tokenwise objectives ignore how this future mismatch depends on the first decision, providing zero gradient and getting stuck at a suboptimal solution, while sequence-level optimization correctly propagates this signal.

Taken together, these results highlight two distinct issues with existing self-distillation approaches: the choice of objective can fail to guarantee monotonic improvement, and the associated tokenwise gradient estimators fail to assign credit to early decisions. Addressing these failures requires reconsidering both the objective and how gradients are computed. We turn to this next.

4. RL from Rich Feedback with Distributional Imitation Learning

Motivated by the limitations of existing self-distillation approaches, we seek an objective that ensures consistent policy improvement, along with a training procedure that enables proper credit assignment across the sequence.

4.1. Distributional Imitation Learning Formulation

We view self-distillation through the lens of *on-policy imitation learning*. Our setting closely resembles DAgger (Ross et al., 2011): the learner collects data under its own policy, queries an expert on visited states, and updates via imitation. This perspective is particularly natural for LLMs, where the

student generates trajectories and the teacher can be queried for supervision at each visited state. A key departure from classical DAgger is our use of a *distributional objective*: rather than supervising individual actions, we minimize a cross-entropy loss between student and teacher distributions. Crucially, this objective does not require access to the full teacher distribution: it admits unbiased estimation from a single sample, can leverage a small set of samples (e.g., top-k), or utilize the full distribution when available. This flexibility allows our method to operate seamlessly across both black-box teachers and fully accessible models. Formally, we consider the $\mathcal{L}_{\text{DistIL}}(\theta)$ objective as follows

$$\mathbb{E}_{\substack{x\sim\rho \\ y\sim\pi_\theta(\cdot|x)}}\left[\sum_{t=1}^{H} H^{\times}\big(\mathsf{sg}\big(\pi_T(\cdot \mid x, y_{1:t-1})\big), \pi_\theta(\cdot \mid x, y_{1:t-1})\big)\right] \tag{4}$$

where $\pi_T(\cdot \mid x, y_{1:t-1}) = \mathrm{sg}\left(\pi_\theta(\cdot \mid x, y_{1:t-1}, f)\right)$. Our objective has a key advantage over SDPO: it can leverage a black-box teacher (e.g., Claude or human annotators) through sample-based estimation, whereas SDPO requires explicit access to the teacher distribution.

4.2. Local and future credit assignment

Next, we remove the stop-gradient on the sampling process, thereby allowing gradients to correctly reflect the dependence of future outcomes on earlier decisions. We provide the gradient of our $\mathcal{L}_{\text{DistIL}}(\theta)$ in Proposition 4.1.

Proposition 4.1. *For $s_t = (x, y_{1:t-1})$, the gradient of $\mathcal{L}_{\text{DistIL}}(\theta)$ is given by*

$$\underbrace{\mathbb{E}_{y\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta H^{\times}\Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid s_t, f)\big),\ \pi_\theta(\cdot \mid s_t)\Big)\right]}_{\textit{local credit assignment}} + \underbrace{\mathbb{E}_{y\sim\pi_\theta}\left[\sum_{t=1}^{H} g_t \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]}_{\textit{future-credit assignment}}, \tag{5}$$

The second term in Eq. (5) propagates future teacher–student disagreement back to earlier decisions through the score function term $\nabla_\theta \log \pi_\theta(a_t \mid s_t)$. This stands in contrast to local token-wise gradient estimators, such as those used in OPSD and SDPO, which omit this dependence by treating the trajectory distribution as fixed, and thus fail to correctly account for how early decisions influence future outcomes. The derivation of Proposition 5 is provided in appendix B.4.

5. Theoretical Properties of DistIL

We now present theoretical guarantees for DistIL. First, we show DistIL enjoys policy improvement. We subsequently derive a regret bound, and finally connect our objective to maximum-likelihood reinforcement learning (MaxRL), providing additional insight into its behavior.

5.1. Monotonic Policy Improvement

Our analysis relies on a mild tangent space assumption, under which we establish a monotonic improvement property of our objective.

Assumption 5.1 (Tangent space realizability)**.** Assume that the teacher direction lies in the tangent space of the student policy manifold at θ. That is, there exists a vector $u \in \mathbb{R}^d$ such that for all states s and actions a,

$$\log \pi_T(a \mid s) - \log \pi_\theta(a \mid s) = u^\top \nabla_\theta \log \pi_\theta(a \mid s).$$

Intuitively, Assumption 5.1 requires that the discrepancy between the teacher and student policies lies in the span of the student's score functions, meaning that the direction from the student to the teacher is locally realizable by the policy class. This is a mild condition for sufficiently expressive policies. In particular, it holds exactly for tabular softmax policies. A formal statement and proof are provided in Appendix B.5.

Proposition 5.2. *Under Assumption 5.1, suppose the teacher policy is strictly better than the current policy (i.e., $\Delta > 0$). Then, the NPG step minimizing $H^{\times}(\pi_T \| \pi_\theta)$, i.e., $\theta' = \theta - \eta\, F(\theta)^{-1} \nabla_\theta H^{\times}(\pi_T \| \pi_\theta)$ for sufficiently small $\eta > 0$, satisfies $J(\pi_{\theta'}) = J(\pi_\theta) + \eta\Delta + O(\eta^2)$, and hence the update provides monotonic policy improvement.*

Proposition 5.2 shows that, under the tangent space assumption, the natural gradient update aligns the policy with the teacher direction, yielding monotonic improvement whenever the teacher is better. A detailed analysis for frozen state distribution is provided in Appendix B.6.

Remark. In the forward Cross Entropy loss, the NPG step becomes a convex interpolation toward the teacher, so improvement follows directly from the fact that we assume the teacher is superior. However, in reverse KL (SDPO), the NPG step is not an interpolation and has an extra weight $\log \pi_\theta / \pi_T$. This term can lead to increased probability on bad actions, even if the teacher is better on average. Note that this argument goes beyond the typical mode-seeking vs. mean-seeking of forward and reverse KL.

5.2. Regret Guarantee of DistIL

Next, we show that our algorithm enjoys regret bound.

Theorem 5.3. *Let π_S be the student policy returned by the updates in Eq. (4). DistIL ensures that*

$$J(\pi_T) - J(\pi_S) \lesssim \sqrt{\bar{\sigma}^2_{\pi_T} \frac{HC}{2}} \left(\frac{2C_0}{n}\right)^{1/4} + \frac{\mu_T H C}{2} \sqrt{\frac{2C_0}{n}}.$$

Method	Chemistry 1h	Chemistry 5h	Physics 1h	Physics 5h	Biology 1h	Biology 5h	Materials 1h	Materials 5h	Average 1h	Average 5h
Qwen3-8B	41.2		59.2		30.8		58.9		47.5	
+ GRPO (off-policy)	65.9	74.5	63.8	72.7	35.1	59.9	74.3	**77.1**	59.8	71.1
+ GRPO (on-policy)	63.3	63.4	63.6	63.6	49.8	49.8	73.9	74.1	62.7	62.7
+ SDPO	73.0	80.2	68.0	72.4	52.9	63.6	72.2	75.9	66.5	73.0
+ DistIL *(Ours)*	**75.8**	**80.8**	**72.7**	**80.8**	**53.3**	**66.6**	**74.9**	76.2	**69.2**	**76.1**
Olmo3-7B-Instruct	22.8		37.7		16.2		36.7		28.4	
+ GRPO (off-policy)	39.7	56.7	55.3	63.3	35.6	**55.8**	70.9	75.0	50.4	62.7
+ GRPO (on-policy)	51.4	57.5	62.7	62.7	**49.8**	49.8	73.3	73.5	59.3	60.9
+ SDPO	70.2	79.2	59.8	64.9	49.5	52.9	71.8	**78.1**	62.8	68.8
+ DistIL *(Ours)*	**72.1**	**81.0**	**67.4**	**74.5**	47.8	55.3	**73.5**	76.9	**65.2**	**71.9**

Table 1. Comparison on scientific reasoning benchmarks (SciKnowEval L3). We report best avg@16 within 1h and 5h of wall-clock training on 4× NVIDIA H200 GPUs. Average is computed over all four domains. **Bold** denotes the best result per column; underline denotes the second best.

We refer to Appendix B.7 for formal definitions and properties of C, C_0, $\bar{\sigma}_{\pi_T}$ and μ_T, and the proof of Theorem 5.3. Theorem 5.3 shows that the regret of the learned policy is controlled through two problem-dependent quantities: $\bar{\sigma}_{\pi_T}$ and μ_T. The term $\bar{\sigma}_{\pi_T}$ captures the stochasticity of the teacher while μ_T measures the worst-case sensitivity of the environment to taking suboptimal actions. For a nearly deterministic teacher, the second term dominates and yields a faster $O(n^{-1/2})$ rate whereas for highly stochastic teachers, the variance term leads to a slower $O(n^{-1/4})$ rate.

Remark. Our analysis builds on the per-state mirror descent framework of (Chang et al., 2024), but requires a non-trivial extension to the online setting. In particular, their analysis assumes expectations with respect to a fixed distribution, unlike our setting where the state distribution evolves on every iteration. We address this by deriving a recurrence relation for the intermediate error terms, enabling us to extend the analysis to the online regime. Full details are provided in Appendix B.7. This technique may be of independent interest for analyzing other online learning algorithms.

5.3. Connections to MaxRL

Finally, our forward cross-entropy objective can be interpreted as maximizing a lower bound on a teacher-weighted maximum likelihood RL (MaxRL) objective (Tajwar et al., 2026). In this regard, we have the following proposition.

Proposition 5.4. *For a fixed prompt x, define the success probability $p_\pi(x) := \mathbb{E}_{y\sim\pi(\cdot|x)}[r(x,y)]$, where $r(x,y) \in \{0,1\}$ indicates correctness. Then, $-H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) \le p_{\pi_T}(x) \log p_{\pi_\theta}(x)$.*

Remark. Proposition 5.4 shows that minimizing forward cross-entropy maximizes a lower bound on a teacher-weighted success objective, linking our DistIL to MaxRL-style optimization. The result follows by partitioning the cross-entropy over correct and incorrect responses and lower bounding the contribution of correct responses, while noting that the remaining terms are nonnegative. Thus, optimizing cross-entropy implicitly maximizes a teacher-weighted log-likelihood of success. This provides a principled explanation for our improved Pass@n performance, as MaxRL objectives are shown to optimize Pass@n for all n (Tajwar et al., 2026). More discussion is provided in Appendix B.8.

6. Experimental Results

We evaluate our method across three representative settings that capture different feedback regimes and levels of supervision: (i) science reasoning benchmarks without rich environment feedback; (ii) coding tasks with rich execution-based feedback; and (iii) mathematical reasoning on challenging problems with access to ground-truth solutions.

6.1. Learning under Sparse Feedback: Science

We evaluate in the standard RLVR setting, where only a binary success signal is observed per rollout. Following SDPO, we sample multiple rollouts per input and use successful trajectories as feedback f, training the model to increase their likelihood and converting sparse rewards into a dense, self-supervised signal.

Setup. We compare against SDPO, on-policy GRPO (trajectories from the current policy), and off-policy GRPO (past rollouts reweighted via importance sampling). For all baselines, we adopt the same hyperparameter settings as in

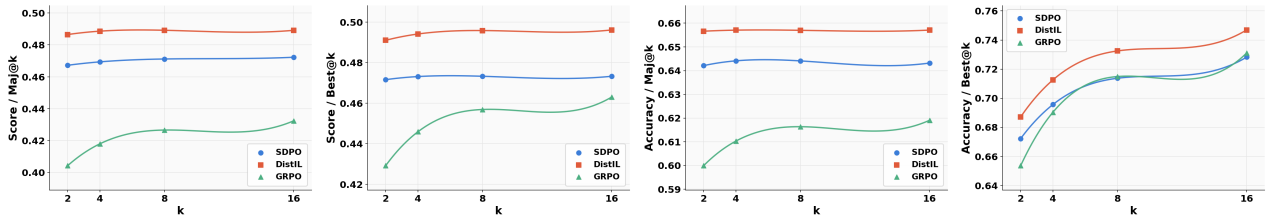

Figure 2. LCBv6 evaluation at τ=0.2 for checkpoint at step-80 (following SDPO), reporting Score and Accuracy at Best@k and Maj@k.

SDPO. Scientific reasoning is assessed on undergraduate-level problems in chemistry, physics, biology, and materials science using the L3 reasoning subset of SciKnowEval (Feng et al., 2024), with a train–test split for in-domain generalization. We initialize from Qwen38B (Yang et al., 2025a) and Olmo3-7B-Instruct (Olmo et al., 2025), and report Best@16 and Maj@16 (Figure 1) and Avg@16 (Table 1) against wall-clock training time using the `verl` framework (Sheng et al., 2025).

Empirical Findings. Table 1 and Figure 1 summarize our results. DistIL achieves the best avg@16 performance in the large majority of columns across both model families, with the largest gains in physics and chemistry domains requiring multi-level reasoning, where sequence-level credit assignment (Proposition 4.1) is most impactful; physics gains reach 8.1/9.6 points at 5h over next-best on Qwen3-8B/OLMo3. Beyond accuracy, DistIL's lead emerges within $\sim$20 steps and is largely monotonically sustained (Proposition 5.2), while SDPO exhibits instability: Best@16 collapses on biology after step 100 and oscillates on chemistry and physics. DistIL's Best@16 advantage is grounded in Proposition 5.4: our forward cross-entropy objective maximizes a teacher-weighted lower bound on success probability, implicitly optimizing `Pass@N` for all N (Tajwar et al., 2026). On-policy GRPO saturates within one hour; off-policy GRPO lags DistIL by 5+ points at 5h.

6.2. Learning with Rich Environment Feedback: Code

We evaluate in environments with rich feedback, where signals go beyond binary success, such as execution errors, which standard RLVR cannot exploit. Following SDPO, we sample multiple rollouts per input and use execution-based feedback to construct supervision. Concretely, rollouts are evaluated against public unit tests, and feedback f is derived from the resulting code execution logs.

Setup. We compare against SDPO and GRPO with identical hyperparameters as in SDPO on LiveCodeBench (LCBv6) (Jain et al., 2024), using public tests for training feedback and private tests for evaluation, initialized from Qwen3-8B. Following (Jain et al., 2024), at temperature 0.2, top_p 0.95, we report exact correctness by determining whether all test cases are passed (Score/Avg@k, Score/Maj@k, Score/Best@k) and partial correctness: fraction of test cases passed (Accuracy/Avg@k, Accuracy/Maj@k, Accuracy/Best@k).

Empirical Findings. DistIL achieves Accuracy/Avg@16 = 0.656 and Score/Avg@16 = 0.482, outperforming SDPO (0.643, 0.467) and GRPO (0.600, 0.405). The large gap over GRPO reflects its fundamental inability to exploit execution feedback. Figure 2 further shows consistent gains across all k on both Best@k and Maj@k for score and accuracy, with the advantage most pronounced at small k. The gap between DistIL and SDPO could be attributed to our theoretically grounded-objective and squence-level credit assignment.

6.3. Learning to Solve Very Hard Mathematical Reasoning Problems

We evaluate with access to ground-truth solutions as feedback f on very hard mathematical reasoning problems, enabling learning on problems where the base model never produces a correct rollout. In this regime, RLVR methods such as GRPO receive zero advantage and cannot improve, while distillation-based approaches directly leverage ground-truth solutions to make progress.

Setup. We construct a challenging training set of 738 hard mathematical problems sourced from (CMU AIRe Lab, 2026) and OmniMath (problems with pass@512 = 0 for Qwen3-4B-Instruct), following (Setlur et al., 2026; Qu et al., 2026). We compare against OPSD (forward-KL), SDPO (reverse-KL), GRPO, and SFT, initialized from Qwen3-8B and Qwen3-4B-Instruct-2507 models. Evaluation covers AIME24, AIME25, HMMT25, AMC23, and Minerva.

Empirical Findings. Table 2 summarizes results. First, we observe that, GRPO matches the base model exactly on both Qwen3-4B and Qwen3-8B across all benchmarks, confirming the expected failure mode: on problems where the base model has zero pass rate, GRPO receives zero advantage and cannot improve. DistIL and OPSD/SDPO avoid this collapse by leveraging ground-truth solutions as feedback directly. Second, DistIL achieves the best overall performance across both model scales, with the largest gains on Qwen3-4B where the base model is weakest. On Qwen3-8B, where the base model is already strong, gains are more

Model	Method	AIME24		AIME25		HMMT25		AMC23		Minerva	
		Avg	Pass	Avg	Pass	Avg	Pass	Avg	Pass	Avg	Pass
Qwen3-4B	Base	61.4	82.0	50.3	69.5	30.3	48.8	93.8	98.8	43.2	48.6
	+ SFT	55.8	80.9	43.1	67.4	29.7	46.0	91.7	97.0	41.9	50.4
	+ GRPO	61.4	82.0	50.3	69.5	30.3	48.8	93.8	98.8	43.2	48.6
	+ SDPO	60.9	<u>85.4</u>	49.6	74.1	<u>32.8</u>	50.9	93.9	<u>99.8</u>	<u>43.4</u>	<u>52.0</u>
	+ OPSD	<u>63.2</u>	<u>85.4</u>	<u>51.5</u>	<u>76.1</u>	**33.0**	<u>54.2</u>	<u>94.5</u>	99.6	<u>43.4</u>	51.8
	+ DistIL *(Ours)*	**65.3**	**87.5**	**55.3**	**77.6**	**33.0**	**56.6**	**94.8**	**100.0**	**44.2**	**52.9**
Qwen3-8B	Base	75.9	86.8	66.9	79.6	44.7	65.6	95.6	**100.0**	48.9	56.3
	+ SFT	74.2	85.3	65.3	80.7	42.2	64.4	96.3	**100.0**	48.2	56.2
	+ GRPO	75.9	86.8	66.9	79.6	44.7	65.6	95.6	**100.0**	48.9	56.3
	+ SDPO	**76.9**	89.5	68.3	<u>84.0</u>	<u>45.6</u>	<u>71.2</u>	<u>96.5</u>	**100.0**	<u>49.2</u>	57.7
	+ OPSD	<u>76.5</u>	**91.3**	<u>69.7</u>	83.3	45.5	68.2	96.2	**100.0**	<u>49.2</u>	<u>57.8</u>
	+ DistIL *(Ours)*	76.4	<u>90.7</u>	**71.1**	**85.0**	**46.4**	**71.4**	**96.6**	**100.0**	**49.5**	**58.4**

Table 2. Performance comparison on mathematical reasoning benchmarks for Qwen3 models. **Bold** denotes the best result per column; <u>underline</u> denotes the second best. Following OPSD (Zhao et al., 2026), we report the best checkpoint up to step 100.

modest but consistent across AIME25, HMMT25, and Minerva. Notably, SFT degrades below the base model on most benchmarks, indicating that naive behavioral cloning on ground-truth solutions of hard problems leads to overfitting rather than generalizable reasoning.

An ablation study of DistIL is provided in Appendix E.2.

7. Related Work

Credit Assignment in RLVR. RLVR applies identical trajectory-level signals to all tokens, making credit assignment challenging for long-horizon reasoning (Shao et al., 2024; Guo et al., 2025). Process reward models introduce intermediate supervision but require additional annotation or auxiliary models (Lightman et al., 2023; Wang et al., 2024). Alternative approaches reweight token updates using heuristic internal signals such as entropy or uncertainty (Xie et al., 2025; Li et al., 2026c). In contrast, DistIL constructs token-level supervision directly from privileged feedback, without auxiliary models or heuristics.

On-Policy Distillation and Self-Distillation. On-policy distillation trains a student on its own rollouts under divergence-based teacher supervision, reducing distribution mismatch relative to off-policy approaches (Agarwal et al., 2024; Lu & Lab, 2025). Recent work extends this to self-distillation, where a single model acts as both teacher and student conditioned on privileged feedback (Zhao et al., 2026; Hübotter et al., 2026; Yang et al., 2026a). However, self-distillation can suppress epistemic uncertainty (Kim et al., 2026) and is sensitive to teacher-student alignment (Li et al., 2026b); recent methods address this via competence-aware weighting and hybrid routing (Xu et al., 2026; Li et al., 2026a). Our work revisits the objective: replacing divergence minimization with a forward cross-entropy formulation yields monotonic policy improvement guarantees and stable gradient estimation, connecting self-distillation to DAgger (Ross et al., 2011). Appendix C details related work.

8. Conclusion

We view reinforcement learning from rich feedback as an on-policy imitation learning problem and propose DistIL, a distributional extension of DAgger based on forward cross-entropy. We identify two fundamental limitations of existing self-distillation methods: (i) optimizing any f-divergence provides no monotonic policy improvement guarantee and can actively upweight suboptimal actions; and (ii) token-wise gradient estimation performs only local credit assignment, preventing future outcomes from informing past decisions and causing error accumulation over long sequences. DistIL addresses both by replacing divergence minimization with a forward cross-entropy objective that aligns updates with the teacher advantage direction and propagates credit through the student-induced state distribution. We prove monotonic policy improvement under a mild realizability condition, sublinear regret over training iterates, and a connection to MaxRL that implicitly optimizes Pass@n for all n simultaneously. Empirically, DistIL generally achieves the best performance across most scientific reasoning, coding, and hard mathematical reasoning. Taken together, our results underscore that objective design and gradient structure are consequential for learning from rich feedback, and position distributional imitation learning as a principled and theoretically grounded path toward stronger reasoning in LLMs.

References

Agarwal, R., Vieillard, N., Zhou, Y., Stanczyk, P., Garea, S. R., Geist, M., and Bachem, O. On-policy distillation of language models: Learning from self-generated mistakes. In *The twelfth international conference on learning representations*, 2024.

Ahmadian, A., Cremer, C., Gallé, M., Fadaee, M., Kreutzer, J., Pietquin, O., Üstün, A., and Hooker, S. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, 2024.

Chang, J. D., Zhan, W., Oertell, O., Brantley, K., Misra, D., Lee, J. D., and Sun, W. Dataset reset policy optimization for rlhf. *arXiv preprint arXiv:2404.08495*, 2024.

Chen, G., Liao, M., Li, C., and Fan, K. Step-level value preference optimization for mathematical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7889–7903, 2024.

Chen, M., Chen, G., Wang, W., and Yang, Y. Seed-grpo: Semantic entropy enhanced grpo for uncertainty-aware policy optimization. *arXiv preprint arXiv:2505.12346*, 2025a.

Chen, X., Li, X., Sun, Z., and Yu, W. Beyond high-entropy exploration: Correctness-aware low-entropy segment-based advantage shaping for reasoning llms. *arXiv preprint arXiv:2512.00908*, 2025b.

Cheng, D., Huang, S., Zhu, X., Dai, B., Zhao, X., Zhang, Z., and Wei, F. Reasoning with exploration: An entropy perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 30377–30385, 2026.

CMU AIRe Lab. Pope-hard-w-oracle-solution, 2026. URL https://huggingface.co/datasets/CMU-AIRe/POPE-HARD-w-oracle-solution. Part of the POPE (Privileged On-Policy Exploration) dataset collection.

Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Cui, G., Yuan, L., Wang, Z., Wang, H., Zhang, Y., Chen, J., Li, W., He, B., Fan, Y., Yu, T., et al. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*, 2025.

Dai, M., Yang, C., and Si, Q. S-grpo: Early exit via reinforcement learning in reasoning models. *arXiv preprint arXiv:2505.07686*, 2025.

Feng, K., Shen, X., Wang, W., Zhuang, X., Tang, Y., Zhang, Q., and Ding, K. Sciknoweval: Evaluating multi-level scientific knowledge of large language models. *arXiv preprint arXiv:2406.09098*, 2024.

Foster, D. J., Block, A., and Misra, D. Is behavior cloning all you need? understanding horizon in imitation learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=8KPyJm4gt5.

Gu, Y., Dong, L., Wei, F., and Huang, M. Minillm: Knowledge distillation of large language models. In *The twelfth international conference on learning representations*, 2024.

Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

He, Y., Kaur, S., Bhaskar, A., Yang, Y., Liu, J., Ri, N., Fowl, L., Panigrahi, A., Chen, D., and Arora, S. Self-distillation zero: Self-revision turns binary rewards into dense supervision. *arXiv preprint arXiv:2604.12002*, 2026.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hübotter, J., Lübeck, F., Behric, L., Baumann, A., Bagatella, M., Marta, D., Hakimi, I., Shenfeld, I., Buening, T. K., Guestrin, C., et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Jain, N., Han, K., Gu, A., Li, W.-D., Yan, F., Zhang, T., Wang, S., Solar-Lezama, A., Sen, K., and Stoica, I. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pp. 267–274, 2002.

Kakade, S. M. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

Kim, J., Luo, X., Kim, M., Lee, S., Kim, D., Jeon, J., Li, D., and Yang, Y. Why does self-distillation (sometimes) degrade the reasoning capability of llms? *arXiv preprint arXiv:2603.24472*, 2026.

LAS Group, ETH Zurich. Sdpo: Reinforcement learning via self-distillation. https://github.com/lasgroup/SDPO, 2026.

Li, G., Yang, T., Fang, J., Song, M., Zheng, M., Guo, H., Zhang, D., Wang, J., and Chua, T.-S. Unifying group-relative and self-distillation policy optimization via sample routing. *arXiv preprint arXiv:2604.02288*, 2026a.

Li, Y., Dong, Z., Sun, Y., Wang, W., Xiong, S., Luo, Y., Liu, J., Lu, H., Wang, J., Su, W., et al. Attention illuminates llm reasoning: The preplan-and-anchor rhythm enables fine-grained policy optimization. *arXiv preprint arXiv:2510.13554*, 2025.

Li, Y., Zuo, Y., He, B., Zhang, J., Xiao, C., Qian, C., Yu, T., Gao, H.-a., Yang, W., Liu, Z., et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026b.

Li, Z., Kang, L., Xiao, F., Xing, L., Si, Q., Li, Z., Gong, W., Yang, D., Xiao, Y., and Guo, H. Outcome-grounded advantage reshaping for fine-grained credit assignment in mathematical reasoning. *arXiv preprint arXiv:2601.07408*, 2026c.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

Liu, Z., Chen, C., Li, W., Qi, P., Pang, T., Du, C., Lee, W. S., and Lin, M. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Lu, K. and Lab, T. M. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Luo, L., Liu, Y., Liu, R., Phatale, S., Guo, M., Lara, H., Li, Y., Shu, L., Zhu, Y., Meng, L., et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

Luo, M., Tan, S., Huang, R., Patel, A., Ariyak, A., Wu, Q., Shi, X., Xin, R., Cai, C., Weber, M., et al. Deepcoder: A fully open-source 14b coder at o3-mini level, 2025. Notion Blog.

Olmo, T., Ettinger, A., Bertsch, A., Kuehl, B., Graham, D., Heineman, D., Groeneveld, D., Brahman, F., Timbers, F., Ivison, H., et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

Pan, Z. and Liu, H. Metaspatial: Reinforcing 3d spatial reasoning in vlms for the metaverse. *arXiv preprint arXiv:2503.18470*, 2025.

Qu, Y., Setlur, A., Smith, V., Salakhutdinov, R., and Kumar, A. Pope: Learning to reason on hard problems via privileged on-policy exploration. *arXiv preprint arXiv:2601.18779*, 2026.

Ross, S., Gordon, G., and Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Setlur, A., Wang, Z., Cohen, A., Rashidinejad, P., and Xie, S. M. Reuse your flops: Scaling rl on hard problems by conditioning on very off-policy prefixes. *arXiv preprint arXiv:2601.18795*, 2026.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Sheng, G., Zhang, C., Ye, Z., Wu, X., Zhang, W., Zhang, R., Peng, Y., Lin, H., and Wu, C. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Sun, W., Yang, W., Jian, P., Du, Q., Cui, F., Ren, S., and Zhang, J. Ktae: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning. *arXiv preprint arXiv:2505.16826*, 2025.

Tajwar, F., Zeng, G., Zhou, Y., Song, Y., Arora, D., Jiang, Y., Schneider, J., Salakhutdinov, R., Feng, H., and Zanette, A. Maximum likelihood reinforcement learning. *arXiv preprint arXiv:2602.02710*, 2026.

Tan, S., Luo, M., Wong, J., Cai, C., Shi, X., Tang, W. Y., Roongta, M., Zhang, T., Li, L. E., Popa, R. A., et al.

Deepscaler: Effective rl scaling of reasoning models via iterative context lengthening. *Notion Blog*, 2025.

Wang, P., Li, L., Shao, Z., Xu, R., Dai, D., Li, Y., Chen, D., Wu, Y., and Sui, Z. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, 2024.

Wang, S., Yu, L., Gao, C., Zheng, C., Liu, S., Lu, R., Dang, K., Chen, X., Yang, J., Zhang, Z., et al. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*, 2025.

Xiao, B., Xia, B., Yang, B., Gao, B., Shen, B., Zhang, C., He, C., Lou, C., Luo, F., Wang, G., et al. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

Xie, C., Pan, R., Wu, X., Zhang, Y., Fu, J., Gao, T., and Zhou, G. Unlocking exploration in rlvr: Uncertainty-aware advantage shaping for deeper reasoning. *arXiv preprint arXiv:2510.10649*, 2025.

Xu, W., Han, R., Wang, Z., Le, L. T., Madeka, D., Li, L., Wang, W. Y., Agarwal, R., Lee, C.-Y., and Pfister, T. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. *arXiv preprint arXiv:2410.11325*, 2024.

Xu, Y., Sang, H., Zhou, Z., He, R., and Wang, Z. Paced: Distillation and on-policy self-distillation at the frontier of student competence. *arXiv preprint arXiv:2603.11178*, 2026.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Yang, C., Si, Q., Duan, Y., Zhu, Z., Zhu, C., Li, Q., Chen, M., Lin, Z., and Wang, W. Dynamic early exit in reasoning models. *arXiv preprint arXiv:2504.15895*, 2025b.

Yang, C., Qin, C., Si, Q., Chen, M., Gu, N., Yao, D., Lin, Z., Wang, W., Wang, J., and Duan, N. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026a.

Yang, C., Si, Q., Dai, M., Yao, D., Zheng, M., Chen, M., Lin, Z., and Wang, W. Test-time prompt intervention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 34223–34231, 2026b.

Yue, Y., Yuan, Y., Yu, Q., Zuo, X., Zhu, R., Xu, W., Chen, J., Wang, C., Fan, T., Du, Z., et al. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025.

Zeng, A., Lv, X., Hou, Z., Du, Z., Zheng, Q., Chen, B., Yin, D., Ge, C., Huang, C., Xie, C., et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Zeng, W., Huang, Y., Liu, Q., Liu, W., He, K., Ma, Z., and He, J. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*, 2025.

Zhang, L., Hosseini, A., Bansal, H., Kazemi, M., Kumar, A., and Agarwal, R. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*, 2024.

Zhang, Z., Zheng, C., Wu, Y., Zhang, B., Lin, R., Yu, B., Liu, D., Zhou, J., and Lin, J. The lessons of developing process reward models in mathematical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 10495–10516, 2025.

Zhao, S. Opsd: On-policy self-distillation for large language models. https://github.com/siyan-zhao/OPSD, 2026.

Zhao, S., Xie, Z., Liu, M., Huang, J., Pang, G., Chen, F., and Grover, A. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Appendices

A. Useful Lemmas

Lemma A.1 (Fisher identity for softmax policies)**.** *Let π_θ be a softmax policy over a finite action set $\mathcal{A}$, with*

$$\pi_\theta(a) = \frac{e^{\theta_a}}{\sum_{b\in\mathcal{A}} e^{\theta_b}}.$$

Define the score vector $s_a := \nabla_\theta \log \pi_\theta(a)$, and the Fisher information matrix $F(\theta) = \mathbb{E}_{a\sim\pi_\theta}\left[s_a s_a^\top\right]$. Then, for any $a, b \in \mathcal{A}$,

$$s_a^\top F(\theta)^{-1} s_b = \frac{\mathbf{1}\{a=b\}}{\pi_\theta(b)} - 1.$$

Equivalently, $\pi_\theta(a)\, s_a^\top F(\theta)^{-1} s_b = \mathbf{1}\{a=b\} - \pi_\theta(a)$.

Proof. Let $m := |\mathcal{A}|$ and we write $p_a := \pi_\theta(a)$. Let $e_a \in \mathbb{R}^m$ denote the a-th standard basis vector, and let $\mathbf{1} \in \mathbb{R}^m$ denote the all-ones vector. Then, for the softmax parameterization, we have

$$\frac{\partial \log \pi_\theta(a)}{\partial \theta_c} = \mathbf{1}\{a=c\} - p_c.$$

Hence, $s_a = e_a - p$, where $p = (p_1, \ldots, p_m)^\top$.

Now, by definition, $F(\theta) = \mathbb{E}_{a\sim p}\left[(e_a - p)(e_a - p)^\top\right]$. Upon expanding, we get $F(\theta) = \mathbb{E}_{a\sim p}[e_a e_a^\top] - pp^\top$. Since $\mathbb{E}_{a\sim p}[e_a e_a^\top] = \mathrm{diag}(p)$, we obtain

$$F(\theta) = \mathrm{diag}(p) - pp^\top.$$

Next, we fix $b \in \mathcal{A}$, and define $v := \frac{e_b}{p_b} - \mathbf{1}$. We claim that $F(\theta)v = s_b$. For this, we note that $p^\top v = \sum_{a\in\mathcal{A}} p_a \left(\frac{\mathbf{1}\{a=b\}}{p_b} - 1\right) = 1 - 1 = 0$. Therefore,

$$F(\theta)v = (\mathrm{diag}(p) - pp^\top)v = \mathrm{diag}(p)v - p(p^\top v) = \mathrm{diag}(p)v.$$

Now,

$$\mathrm{diag}(p)v = \mathrm{diag}(p)\left(\frac{e_b}{p_b} - \mathbf{1}\right) = e_b - p = s_b.$$

Thus, $F(\theta)v = s_b$. Now, assuming that inverse of $F(\theta)$ exists, we have

$$F(\theta)^{-1} s_b = v = \frac{e_b}{p_b} - \mathbf{1}.$$

Finally, using $(s_a = e_a - p)$, we have $s_a^\top F(\theta)^{-1} s_b = s_a^\top \left(\frac{e_b}{p_b} - \mathbf{1}\right)$. Since $s_a^\top \mathbf{1} = 0$, this simplifies to

$$s_a^\top F(\theta)^{-1} s_b = \frac{s_a^\top e_b}{p_b} = \frac{\mathbf{1}\{a=b\} - p_b}{p_b} = \frac{\mathbf{1}\{a=b\}}{p_b} - 1.$$

Moreover, multiplying both sides by (p_a) gives the equivalent form

$$p_a\, s_a^\top F(\theta)^{-1} s_b = \mathbf{1}\{a=b\} - p_a.$$

□

Lemma A.2 (Trajectory score-function identity)**.** *Let $\tau \sim p_\theta(\tau)$ be a differentiable trajectory distribution, and let $f(\tau, \theta)$ be any differentiable function. Then*

$$\nabla_\theta \mathbb{E}_{\tau\sim p_\theta}[f(\tau,\theta)] = \mathbb{E}_{\tau\sim p_\theta}\left[f(\tau,\theta)\nabla_\theta \log p_\theta(\tau) + \nabla_\theta f(\tau,\theta)\right].$$

In particular, if f does not depend on θ, then $\nabla_\theta \mathbb{E}_{\tau\sim p_\theta}[f(\tau)] = \mathbb{E}_{\tau\sim p_\theta}\left[f(\tau)\nabla_\theta \log p_\theta(\tau)\right]$. Moreover,

$$\mathbb{E}_{\tau\sim p_\theta}\left[\nabla_\theta \log p_\theta(\tau)\right] = 0.$$

Proof. Using the definition of expectation and differentiating it gives,

$$\nabla_\theta \mathbb{E}_{\tau\sim p_\theta}[f(\tau,\theta)] = \nabla_\theta \sum_\tau p_\theta(\tau) f(\tau,\theta) = \sum_\tau \nabla_\theta p_\theta(\tau)\, f(\tau,\theta) + \sum_\tau p_\theta(\tau)\nabla_\theta f(\tau,\theta).$$

Next, we use $\nabla_\theta p_\theta(\tau) = p_\theta(\tau)\nabla_\theta \log p_\theta(\tau)$, to get

$$\begin{aligned}\nabla_\theta \mathbb{E}_{\tau\sim p_\theta}[f(\tau,\theta)] &= \sum_\tau p_\theta(\tau) f(\tau,\theta)\nabla_\theta \log p_\theta(\tau) + \sum_\tau p_\theta(\tau)\nabla_\theta f(\tau,\theta)\\ &= \mathbb{E}_{\tau\sim p_\theta}\big[f(\tau,\theta)\nabla_\theta \log p_\theta(\tau) + \nabla_\theta f(\tau,\theta)\big],\end{aligned}$$

Now, if f does not depend on θ, the second term vanishes. Finally,

$$\mathbb{E}_{\tau\sim p_\theta}\big[\nabla_\theta \log p_\theta(\tau)\big] = \sum_\tau p_\theta(\tau)\nabla_\theta \log p_\theta(\tau) = \sum_\tau \nabla_\theta p_\theta(\tau) = \nabla_\theta \sum_\tau p_\theta(\tau) = \nabla_\theta 1 = 0.$$

□

Lemma A.3. *Let $\Phi : \Pi \to \mathbb{R}$ be a differentiable strictly convex function, and let*

$$D_\Phi(\pi\|\rho) := \Phi(\pi) - \Phi(\rho) - \langle \nabla\Phi(\rho),\, \pi - \rho\rangle$$

denote the Bregman divergence induced by Φ. For the following mirror-descent trust-region update:

$$\pi^{i+1} \in \arg\min_{\pi\in\Pi}\left\{\langle \nabla \ell_i(\pi^i), \pi\rangle + \frac{1}{\eta} D_\Phi(\pi\|\pi^i)\right\},$$

we have, for all $\pi \in \Pi$,

$$\big\langle \eta\nabla\ell_i(\pi^i) + \nabla\Phi(\pi^{i+1}) - \nabla\Phi(\pi^i),\, \pi - \pi^{i+1}\big\rangle \geq 0.$$

Proof. The objective is convex in π, and π^{i+1} is a minimizer over the convex set Π. Therefore, the standard first-order optimality condition gives

$$\big\langle \nabla\ell_i(\pi^i) + \frac{1}{\eta}\big(\nabla\Phi(\pi^{i+1}) - \nabla\Phi(\pi^i)\big),\, \pi - \pi^{i+1}\big\rangle \geq 0,$$

for all $\pi \in \Pi$. Multiplying by η gives the final inequality. □

Lemma A.4. *For any differentiable strictly convex Φ and any u, v, w in its domain,*

$$\big\langle \nabla\Phi(u) - \nabla\Phi(v),\, w - u\big\rangle = D_\Phi(w,v) - D_\Phi(w,u) - D_\Phi(u,v).$$

Proof. By definition,

$$D_\Phi(x,y) = \Phi(x) - \Phi(y) - \langle \nabla\Phi(y), x - y\rangle.$$

Upon expanding $D_\Phi(w,v) - D_\Phi(w,u) - D_\Phi(u,v)$ and rearranging terms gives

$$\Phi(w) - \Phi(v) - \langle \nabla\Phi(v), w - v\rangle - \Phi(w) + \Phi(u) + \langle \nabla\Phi(u), w - u\rangle - \Phi(u) + \Phi(v) + \langle \nabla\Phi(v), u - v\rangle,$$

which simplifies to

$$\langle \nabla\Phi(u) - \nabla\Phi(v),\, w - u\rangle.$$

□

Lemma A.5. *Let $\{a_i\}_{i\geq 1}$ be a sequence satisfying $a_{i+1} \leq (1-\eta)a_i + b$, where $\eta \in (0,1)$. Then, for all $i \geq 1$,*

$$a_i \leq (1-\eta)^{i-1} a_1 + b\sum_{k=0}^{i-2}(1-\eta)^k.$$

Proof. We prove by induction on i. For $i = 1$, the bound reduces to $a_1 \leq a_1$, which holds trivially. Next, assume the statement holds for some $i \geq 1$. Using the recursion and the induction hypothesis,

$$\begin{aligned} a_{i+1} &\leq (1-\eta)\left[(1-\eta)^{i-1}a_1 + b\sum_{k=0}^{i-2}(1-\eta)^k\right] + b \\ &= (1-\eta)^i a_1 + b\sum_{k=0}^{i-2}(1-\eta)^{k+1} + b \\ &= (1-\eta)^i a_1 + b\sum_{k=0}^{i-1}(1-\eta)^k. \end{aligned}$$

This establishes the claim for $i + 1$ and completes the proof. □

B. Theoretical Results

B.1. Lack of Monotonic Improvement Guarantees under f-Divergences

Proposition 1. Let π_T be a fixed teacher and π_θ a student policy. For the softmax parameterization, the Natural Policy Gradient (NPG) step minimizing an f-divergence $D_f(\pi_T\|\pi_\theta)$, i.e., $\theta' = \theta - \eta\, F(\theta)^{-1}\nabla_\theta D_f(\pi_T\|\pi_\theta)$, satisfies

$$J(\pi_{\theta'}) = J(\pi_\theta) - \eta\, \mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\left(r(y), g(y)\right) + O(\eta^2),$$

where $g(y) = f\left(\frac{\pi_T(y)}{\pi_\theta(y)}\right) - \frac{\pi_T(y)}{\pi_\theta(y)}\, f'\left(\frac{\pi_T(y)}{\pi_\theta(y)}\right)$. Consequently, for sufficiently small $\eta > 0$, the NPG step provides a policy improvement if and only if $\mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\left(r(y), g(y)\right) < 0$. In particular, $\Delta > 0$ does not suffice to guarantee this condition.

Proof. Let us define $p_a := \pi_\theta(a)$, $q_a := \pi_T(a)$, and $u_a := \frac{q_a}{p_a}$. Then, we have

$$\mathcal{L}(\theta) := D_f(\pi_T\|\pi_\theta) = \sum_{a\in\mathcal{A}} p_a\, f(u_a).$$

Next, we compute the gradient of $\mathcal{L}(\theta)$. By the chain rule,

$$\frac{\partial L(\theta)}{\partial p_a} = \frac{\partial}{\partial p_a}\left[p_a f\left(\frac{q_a}{p_a}\right)\right] = f(u_a) - u_a f'(u_a) = g(a),$$

where we used the definition of $g(a)$ from the statement in the Proposition. Next, we define the score vector $s_a := \nabla_\theta \log \pi_\theta(a)$. Using $\nabla_\theta p_a = p_a s_a$ and linearity, we have

$$\nabla_\theta \mathcal{L}(\theta) = \sum_{b\in\mathcal{A}} \frac{\partial \mathcal{L}(\theta)}{\partial p_b}\, \nabla_\theta p_b = \sum_{b\in\mathcal{A}} g(b)\, p_b\, s_b = \mathbb{E}_{b\sim\pi_\theta}[g(b)s_b]\,.$$

Now, the natural gradient update is given as $\theta' - \theta = -\eta F(\theta)^{-1}\nabla_\theta\mathcal{L}(\theta)$. Upon expanding the policy map to first order gives

$$\pi_{\theta'}(a) = \pi_\theta(a) + \nabla_\theta\pi_\theta(a)^\top(\theta' - \theta) + O(\eta^2).$$

Since $\nabla_\theta\pi_\theta(a) = \pi_\theta(a)s_a$ and using the gradient update, we get

$$\pi_{\theta'}(a) = \pi_\theta(a) - \eta\, \pi_\theta(a)\, s_a^\top F(\theta)^{-1}\nabla_\theta\mathcal{L}(\theta) + O(\eta^2).$$

Now, we substitute the expression for $\nabla_\theta\mathcal{L}(\theta)$ to get $s_a^\top F(\theta)^{-1}\nabla_\theta\mathcal{L}(\theta) = \mathbb{E}_{b\sim\pi_\theta}\left[g(b)\, s_a^\top F(\theta)^{-1}s_b\right]$. By Lemma A.1,

$$s_a^\top F(\theta)^{-1}s_b = \frac{\mathbf{1}\{a = b\}}{\pi_\theta(b)} - 1.$$

Therefore,

$$\begin{aligned} s_a^\top F(\theta)^{-1}\nabla_\theta \mathcal{L}(\theta) &= \mathbb{E}_{b\sim\pi_\theta}\left[g(b)\left(\frac{\mathbf{1}\{a=b\}}{\pi_\theta(b)} - 1\right)\right] \\ &= \mathbb{E}_{b\sim\pi_\theta}\left[g(b)\frac{\mathbf{1}\{a=b\}}{\pi_\theta(b)}\right] - \mathbb{E}_{b\sim\pi_\theta}[g(b)] \\ &= g(a) - \mathbb{E}_{b\sim\pi_\theta}[g(b)]. \end{aligned}$$

Thus, we have

$$\pi_{\theta'}(a) = \pi_\theta(a) - \eta\,\pi_\theta(a)\Big(g(a) - \mathbb{E}_{b\sim\pi_\theta}[g(b)]\Big) + O(\eta^2).$$

Next, we consider that change in value of the policy following gradient update. Upon using the definition of $J(\pi)$, we have

$$\begin{aligned} J(\pi_{\theta'}) - J(\pi_\theta) &= \sum_{a\in\mathcal{A}} r(a)\big(\pi_{\theta'}(a) - \pi_\theta(a)\big) \\ &= -\eta \sum_{a\in\mathcal{A}} \pi_\theta(a)\, r(a)\Big(g(a) - \mathbb{E}_{b\sim\pi_\theta}[g(b)]\Big) + O(\eta^2) \\ &= -\eta\left(\mathbb{E}_{a\sim\pi_\theta}[r(a)g(a)] - \mathbb{E}_{a\sim\pi_\theta}[r(a)]\,\mathbb{E}_{a\sim\pi_\theta}[g(a)]\right) + O(\eta^2) \\ &= -\eta\,\mathbb{C}\mathrm{ov}_{a\sim\pi_\theta}\big(r(a), g(a)\big) + O(\eta^2). \end{aligned}$$

Hence, we have $J(\pi_{\theta'}) - J(\pi_\theta) - \eta\,\mathbb{C}\mathrm{ov}_{y\sim\pi_\theta}\big(r(y), g(y)\big) + O(\eta^2)$. □

B.2. Existence of Non-Monotonic Improvement Cases for Reverse-KL

Proposition 2. There exists a contextual bandit setting such that, even when the teacher policy is strictly better than the student policy (i.e., $\Delta > 0$), an NPG step minimizing the reverse KL divergence, as in SDPO, does not lead to policy improvement.

Proof. We provide our construction as follows. We consider a single-context multi-armed bandit with three actions and rewards given as

$$r(1) = 1, \quad r(2) = 1, \quad r(3) = 0.$$

Let the student and teacher policies be

$$\pi_\theta = (0.05,\, 0.10,\, 0.85), \qquad \pi = (0.15,\, 0.05,\, 0.80).$$

Then, we have

$$J(\pi_\theta) = 0.15, \qquad J(\pi) = 0.20,$$

which implies $\Delta > 0$.

Next, we consider the NPG update for reverse-KL which is given as

$$\pi_{\theta'}(y) = \pi_\theta(y) - \eta\,\pi_\theta(y)\left(\log\frac{\pi_\theta(y)}{\pi(y)} - \mathrm{KL}(\pi_\theta\|\pi)\right).$$

Let us now focus on the suboptimal action $y = 3$. We compute

$$\log\frac{\pi_\theta(3)}{\pi(3)} \approx 0.0606, \qquad \mathrm{KL}(\pi_\theta\|\pi) \approx 0.0659,$$

This gives

$$\log\frac{\pi_\theta(3)}{\pi(3)} - \mathrm{KL} \approx -0.0053.$$

Thus,

$$\pi_{\theta'}(3) = \pi_\theta(3) - \eta\cdot(-0.0045) > \pi_\theta(3).$$

Therefore, the probability of the suboptimal action increases after the update, leading to a degradation in performance. Hence, even when $\Delta > 0$, reverse KL NPG does not guarantee monotonic improvement. □

B.3. Local Credit Assignment leads to Sub-Optimal Policies

To show that Proposition 3.3 holds, we provide a detailed construction in Proposition B.1.

Proposition B.1. *Let us consider the action-space $\mathcal{A} = \{a, b, c\}$ and consider a two-step decision process $(y_1, y_2) \in \mathcal{A}^2$. We define the student policy $\pi_{u,v}$ by*

$$\begin{aligned}
\pi_{u,v}(y_1 = \cdot) &= \mathrm{softmax}(u, 0, 0), \\
\pi_{u,v}(y_2 = \cdot \mid y_1 = a) &= \mathrm{softmax}(v, 0, 0), \\
\pi_{u,v}(y_2 = \cdot \mid y_1 = b) &= U, \\
\pi_{u,v}(y_2 = \cdot \mid y_1 = c) &= U,
\end{aligned}$$

where U is the uniform distribution over actions. Let us define the teacher policy π_T as

$$\pi_T(y_1) = U, \qquad \pi_T(\cdot \mid y_1 = a) = R, \qquad \pi_T(\cdot \mid y_1 = b) = \pi_T(\cdot \mid y_1 = c) = U,$$

where R is any distribution satisfying $\inf_v \mathrm{KL}(\mathrm{softmax}(v, 0, 0) \| R) = D_{\min} > 0$. Next, we define the sequence-level objective

$$\mathcal{L}_{\mathrm{seq}}(u, v) = \mathrm{KL}(\pi_{u,v}(y_1, y_2) \| \pi_T(y_1, y_2)).$$

Let $\mathcal{L}_{\mathrm{curr}}$ denote the token-wise objective that sums per-step KL terms but treats the prefix distribution as fixed when differentiating (for example, in SDPO/OPSD).

Then, we have

1. *At $(u, v) = (0, 0)$,*
$$\frac{\partial \mathcal{L}_{\mathrm{curr}}}{\partial u}(0, 0) = 0, \qquad \frac{\partial \mathcal{L}_{\mathrm{seq}}}{\partial u}(0, 0) > 0.$$
2. *Token-wise updates leaves $u = 0$. At the resulting solution v^*, the true objective satisfies*
$$\mathcal{L}_{\mathrm{seq}}(0, v^*) = \frac{1}{3} D_{\min} > 0.$$
3. *In contrast,*
$$\inf_{u,v} \mathcal{L}_{\mathrm{seq}}(u, v) = 0.$$

Thus, token-wise gradients provide no learning signal for the first-step decision and lead to a strictly suboptimal solution, whereas sequence-level optimization admits strictly better solutions.

Proof. We start by using the chain rule for KL divergence which gives

$$\mathcal{L}_{\mathrm{seq}}(u, v) = \mathrm{KL}(\pi_{u,v}(y_1) \| U) + \mathbb{E}_{y_1 \sim \pi_{u,v}} \mathrm{KL}(\pi_{u,v}(y_2 \mid y_1) \| \pi_T(y_2 \mid y_1)).$$

Let us define $p(u) := \pi_{u,v}(y_1 = a) = \frac{e^u}{e^u + 2}$. Then, we have

$$\pi_{u,v}(y_1) = \left(p(u), \frac{1 - p(u)}{2}, \frac{1 - p(u)}{2} \right).$$

Upon simplification, we get

$$\mathrm{KL}(\pi_{u,v}(y_1) \| U) = K(p(u)),$$

where

$$K(p) = p \log(3p) + (1 - p) \log\left(\frac{3(1 - p)}{2} \right).$$

Next, for $y_1 \in \{b, c\}$, both student and teacher distributions are U, so the KL divergence is zero. For $y_1 = a$, let us denote the KL divergence as $D(v)$, where $D(v) = \mathrm{KL}(\mathrm{softmax}(v, 0, 0) \| R)$. Hence, we have $\mathbb{E}_{y_1 \sim \pi_{u,v}} \mathrm{KL}(\pi_{u,v}(y_2 \mid y_1) \| \pi_T(y_2 \mid y_1)) = \mathbb{E}_{y_1} \mathrm{KL}(\cdot) = p(u)D(v)$, and therefore

$$\mathcal{L}_{\mathrm{seq}}(u, v) = K(p(u)) + p(u)D(v).$$

Next, we compute the gradient of sequence-level objective. Upon calculation, we have

$$K'(p) = \log \frac{2p}{1-p}, \qquad p'(u) = \frac{2e^u}{(e^u + 2)^2}.$$

At $u = 0$, we have $p(0) = 1/3$, so

$$K'(1/3) = \log 1 = 0, \qquad p'(0) = \frac{2}{9}.$$

Therefore, we have

$$\frac{\partial \mathcal{L}_{\mathrm{seq}}}{\partial u}(0, 0) = K'(1/3)p'(0) + p'(0)D(0) = \frac{2}{9}D(0) > 0.$$

Next, we focus on the token-wise objective that treats $p(u)$ as fixed when differentiating the second term,

$$\frac{\partial \mathcal{L}_{\mathrm{curr}}}{\partial u} = \frac{d}{du} K(p(u)).$$

Thus, we get

$$\frac{\partial \mathcal{L}_{\mathrm{curr}}}{\partial u}(0, 0) = 0.$$

Therefore, token-wise updates keep $u = 0$ and minimize $D(v)$, giving $D(v^*) = D_{\min}$. Hence,

$$\mathcal{L}_{\mathrm{seq}}(0, v^*) = \frac{1}{3} D_{\min}.$$

On the other hand, as $u \to -\infty$, we have $p(u) \to 0$, and therefore

$$L_{\mathrm{seq}}(u, v) \to 0.$$

To conclude, token-wise optimization is stuck at a strictly suboptimal value, while sequence-level optimization can achieve arbitrarily small loss. □

B.4. DistIL's Gradient Derivation

Proposition 4. The gradient of $\mathcal{L}_{\mathrm{DistIL}}(\theta)$ is given by The gradient of $\mathcal{L}_{\mathrm{DistIL}}(\theta)$ is given by

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\mathrm{DistIL}} =& \mathbb{E}_{y \sim \pi_\theta} \left[\sum_{t=1}^{H} \nabla_\theta H^\times \Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid s_t, f)\big),\ \pi_\theta(\cdot \mid s_t) \Big) \right] \\ &+ \mathbb{E}_{y \sim \pi_\theta} \left[\sum_{t=1}^{H} \nabla_\theta \log \pi_\theta(a_t \mid s_t) \left(\sum_{i > t} H^\times \Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid s_i, f)\big),\ \pi_\theta(\cdot \mid s_i) \Big) \right) \right],\end{aligned}$$

where $s_t = (x, y_{1:t-1})$.

Proof. We start by defining the per-step loss

$$\ell_t(y, \theta) := H^\times \Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid x, y_{1:t-1}, f)\big),\ \pi_\theta(\cdot \mid x, y_{1:t-1}) \Big),$$

so that

$$\mathcal{L}_{\mathrm{DistIL}}(\theta) = \mathbb{E}_{y \sim \pi_\theta} \left[\sum_{t=1}^{H} \ell_t(y, \theta) \right].$$

By Lemma A.2, applied to the trajectory distribution $y \sim \pi_\theta$, we have

$$\nabla_\theta \mathcal{L}_{\text{DistIL}}(\theta) = \mathbb{E}_{y\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta \ell_t(y,\theta)\right] + \mathbb{E}_{\tau\sim\pi_\theta}\left[\left(\sum_{t=1}^{H} \ell_t(\tau,\theta)\right) \nabla_\theta \log \pi_\theta(\tau)\right]. \tag{6}$$

Since the policy is autoregressive, we have $\log \pi_\theta(\tau) = \sum_{t=1}^{H} \log \pi_\theta(a_t \mid s_t)$ and $\nabla_\theta \log \pi_\theta(\tau) = \sum_{t=1}^{H} \nabla_\theta \log \pi_\theta(a_t \mid s_t)$. Upon using this in Eq. 6 gives

$$\nabla_\theta \mathcal{L}_{\text{DistIL}}(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta \ell_t(\tau,\theta)\right] + \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{H}\sum_{i=1}^{H} \ell_i(\tau,\theta)\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right].$$

We now show that only future terms $i > t$ survive in the second expectation. Let us fix t and consider any $i \le t$. The quantity $\ell_i(\tau,\theta)$ depends only on the prefix up to time i, and therefore is independent of the sampled action a_t once the state s_t is fixed. Consequently,

$$\mathbb{E}_{a_t\sim\pi_\theta(\cdot|s_t)}\Big[\nabla_\theta \log \pi_\theta(a_t \mid s_t)\, \ell_i(\tau,\theta)\Big] = \ell_i(\tau,\theta)\, \mathbb{E}_{a_t\sim\pi_\theta(\cdot|s_t)}\Big[\nabla_\theta \log \pi_\theta(a_t \mid s_t)\Big].$$

By Lemma A.2,

$$\mathbb{E}_{a_t\sim\pi_\theta(\cdot|s_t)}\Big[\nabla_\theta \log \pi_\theta(a_t \mid s_t)\Big] = 0,$$

so the contribution from all $i \le t$ vanishes. Therefore,

$$\mathbb{E}_{\tau\sim\pi_\theta}\left[\left(\sum_{t=1}^{H} \ell_t(\tau,\theta)\right) \nabla_\theta \log \pi_\theta(\tau)\right] = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta \log \pi_\theta(a_t \mid s_t) \left(\sum_{i>t} \ell_i(\tau,\theta)\right)\right].$$

Substituting back the definition of ℓ_i gives

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\text{DistIL}} =& \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta H^{\times}\Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid s_t, f)\big),\ \pi_\theta(\cdot \mid s_t)\Big)\right] \\ &+ \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{H} \nabla_\theta \log \pi_\theta(a_t \mid s_t) \left(\sum_{i>t} H^{\times}\Big(\mathsf{sg}\big(\pi_\theta(\cdot \mid s_i, f)\big),\ \pi_\theta(\cdot \mid s_i)\Big)\right)\right].\end{aligned}$$

□

B.5. Tangent-Space Realizability

Proposition B.2. *Let $\mathcal{S}$ and $\mathcal{A}$ be finite sets, and consider a tabular softmax policy*

$$\pi_\theta(a \mid s) = \frac{\exp(\theta(s,a))}{\sum_{a'\in\mathcal{A}} \exp(\theta(s,a'))}.$$

Then, for any fixed θ and any teacher policy π_T, there exists a vector $u : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ such that for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$,

$$\pi_T(a \mid s) - \pi_\theta(a \mid s) = \pi_\theta(a \mid s)\, \nabla_\theta \log \pi_\theta(a \mid s)^\top u.$$

Proof. We start by fixing $s \in \mathcal{S}$. Since $\pi_\theta(\cdot \mid s)$ is a softmax distribution, $\pi_\theta(a \mid s) > 0$ for all $a \in \mathcal{A}$, so the ratio $\pi_T(a \mid s)/\pi_\theta(a \mid s)$ is well-defined. For tabular softmax, the score function satisfies

$$\frac{\partial \log \pi_\theta(a \mid s)}{\partial \theta(s,a')} = \mathbf{1}\{a' = a\} - \pi_\theta(a' \mid s).$$

Thus, for any vector u, we have

$$\nabla_\theta \log \pi_\theta(a \mid s)^\top u = u(s,a) - \sum_{a'\in\mathcal{A}} \pi_\theta(a' \mid s)\, u(s,a').$$

Now, let us define

$$u(s,a) := \frac{\pi_T(a \mid s)}{\pi_\theta(a \mid s)}.$$

Then, we have

$$\sum_{a'} \pi_\theta(a' \mid s)\, u(s,a') = \sum_{a'} \pi_T(a' \mid s) = 1.$$

Hence,

$$\nabla_\theta \log \pi_\theta(a \mid s)^\top u = \frac{\pi_T(a \mid s)}{\pi_\theta(a \mid s)} - 1.$$

Multiplying both sides by $\pi_\theta(a \mid s)$ gives

$$\pi_\theta(a \mid s)\, \nabla_\theta \log \pi_\theta(a \mid s)^\top u = \pi_T(a \mid s) - \pi_\theta(a \mid s).$$

□

B.6. Monotonic Improvement of DistIL

To show that Proposition 5.2 holds, we provide the assumptions made in it, along with the complete proof in Proposition B.3.

Proposition B.3. *Let us consider our objective $\mathcal{L}_{\mathrm{DistIL}}(\theta)$. Assume the following holds*

1. *$\theta \mapsto \pi_\theta$ is twice continuously differentiable,*
2. *the reward $r(x,y)$ is bounded,*
3. *the teacher is strictly better than the student:*

$$\Delta := \mathbb{E}_{x,y\sim\pi_T}[r(x,y)] - \mathbb{E}_{x,y\sim\pi_\theta}[r(x,y)] > 0,$$

4. *(Tangent assumption) there exists $u \in \mathbb{R}^d$ such that for all (x,y),*

$$\pi_T(y \mid x) - \pi_\theta(y \mid x) = \pi_\theta(y \mid x)\, \nabla_\theta \log \pi_\theta(y \mid x)^\top u.$$

Then, the natural gradient update $\theta' = \theta - \eta F(\theta)^{-1} \nabla_\theta \mathcal{L}_{\mathrm{DistIL}}(\theta)$ satisfies

$$\mathbb{E}_{x,y\sim\pi_{\theta'}}[r(x,y)] = \mathbb{E}_{x,y\sim\pi_\theta}[r(x,y)] + \eta\Delta + O(\eta^2).$$

Equivalently, we have $J(\pi_{\theta'}) = J(\pi_\theta) + \eta\Delta + O(\eta^2)$.

Proof. Under the frozen state distribution, the gradient of $\mathcal{L}_{\mathrm{DistIL}}$ is given as

$$\nabla_\theta \mathcal{L}_{\mathrm{DistIL}}(\theta) = -\mathbb{E}_{x\sim\rho,\, y\sim\pi_T(\cdot\mid x)} \left[\nabla_\theta \log \pi_\theta(y \mid x)\right].$$

From Lemma A.2, $\mathbb{E}_{y\sim\pi_\theta(\cdot\mid x)}[\nabla_\theta \log \pi_\theta(y \mid x)] = 0$. Therefore, the gradient can also be written as

$$\nabla_\theta \mathcal{L}_{\mathrm{DistIL}}(\theta) = -\mathbb{E}_{x\sim\rho} \left[\sum_y \left(\pi_T(y \mid x) - \pi_\theta(y \mid x)\right) \nabla \log \pi_\theta(y \mid x)\right].$$

Next, we apply the tangent assumption $\pi_T(y \mid x) - \pi_\theta(y \mid x) = \pi_\theta(y \mid x)\, \nabla_\theta \log \pi_\theta(y \mid x)^\top u$ to obtain

$$\nabla_\theta \mathcal{L}_{\mathrm{DistIL}}(\theta) = -\mathbb{E}_{x\sim\rho} \left[\sum_y \pi_\theta(y \mid x) \nabla_\theta \log \pi_\theta(y \mid x) \nabla_\theta \log \pi_\theta(y \mid x)^\top\right] u.$$

We now recognize the Fisher information matrix,

$$F(\theta) = \mathbb{E}_{x\sim\rho,\, y\sim\pi_\theta(\cdot\mid x)} \left[\nabla_\theta \log \pi_\theta(y \mid x) \nabla_\theta \log \pi_\theta(y \mid x)^\top\right]$$

to conclude

$$\nabla_\theta \mathcal{L}_{\mathrm{DistIL}}(\theta) = -F(\theta)u.$$

Thus, the natural gradient update becomes

$$\theta' = \theta + \eta u.$$

We now analyze the change in the policy using a first-order Taylor expansion

$$\pi_{\theta'}(y \mid x) = \pi_\theta(y \mid x) + \nabla_\theta \pi_\theta(y \mid x)^\top (\theta' - \theta) + O(\eta^2).$$

Substituting $\theta' - \theta = \eta u$ gives

$$\pi_{\theta'}(y \mid x) = \pi_\theta(y \mid x) + \eta \nabla_\theta \pi_\theta(y \mid x)^\top u + O(\eta^2).$$

Using $\nabla_\theta \pi_\theta(y \mid x) = \pi_\theta(y \mid x) \nabla_\theta \log \pi_\theta(y \mid x)$, we obtain

$$\pi_{\theta'}(y \mid x) = \pi_\theta(y \mid x) + \eta \pi_\theta(y \mid x) \nabla_\theta \log \pi_\theta(y \mid x)^\top u + O(\eta^2).$$

Next, we again apply the tangent assumption to obtain

$$\pi_{\theta'}(y \mid x) = \pi_\theta(y \mid x) + \eta\big(\pi_T(y \mid x) - \pi_\theta(y \mid x)\big) + O(\eta^2).$$

Multiplying by $r(x, y)$, summing over y, and taking expectation over $x \sim \rho$, we obtain

$$\mathbb{E}_{x,y\sim\pi_{\theta'}}[r(x,y)] = \mathbb{E}_{x,y\sim\pi_\theta}[r(x,y)] + \eta\left(\mathbb{E}_{x,y\sim\pi_T}[r(x,y)] - \mathbb{E}_{x,y\sim\pi_\theta}[r(x,y)]\right) + O(\eta^2).$$

Finally, using the definition of Δ and $J(\pi)$, this becomes

$$J(\pi_{\theta'}) = J(\pi_\theta) + \eta\Delta + O(\eta^2),$$

Since $\Delta > 0$, for sufficiently small $\eta > 0$, the update gives strict improvement. □

B.7. Regret Analysis

Let us denote $\pi^\star := \pi_T$. We start by assuming that $\left|\frac{\pi^\star(a|s)}{\pi^i(a|s)}\right| \le C, \forall s \in \mathcal{S}, a \in \mathcal{A}$. This is true due to trust-region and also teacher is just feedback conditioned policy.

Theorem B.4. *Let $\ell_i(\pi)$ be defined as*

$$\ell_i(\pi) := -\sum_{t=1}^{H} \mathbb{E}_{s_t\sim d_t^{\pi^i}}\left[\sum_{a\in\mathcal{A}} \pi^\star(a \mid s_t) \log \pi(a \mid s_t)\right].$$

Let us denote $\ell_s(\pi) = -\sum_{a\in\mathcal{A}} \pi^\star(a \mid s) \log \pi(a \mid s)$ for any state $s \in \mathcal{S}$ and $\{\pi^i\}_{i=1}^{n+1}$ be generated by

$$\pi^{i+1}(\cdot \mid s) = \arg\min_{q\in\Delta(\mathcal{A})} \left\{\langle \nabla \ell_s(\pi^i), q\rangle + \frac{1}{\eta} D_{\mathrm{KL}}(q \| \pi^i(\cdot \mid s))\right\}.$$

Under the assumption that

$$\left|\frac{\pi^\star(a|s)}{\pi^i(a|s)}\right| \le C \quad \forall s, a \quad \textit{and} \quad D_{\mathrm{KL}}(\pi^\star(\cdot \mid s) \| \pi^1(\cdot \mid s)) \le C_0, \forall s.$$

we have

$$\frac{1}{n}\sum_{i=1}^{n}\left(\ell_i(\pi^i) - \ell_i(\pi^\star)\right) \le \frac{HC_0}{\eta n} + \frac{\eta H C^2}{2}.$$

Therefore, upon choosing $\eta = \sqrt{\frac{2C_0}{C^2 n}}$*, we obtain*

$$\frac{1}{n}\sum_{i=1}^{n}(\ell_i(\pi^i) - \ell_i(\pi^\star)) \le HC\sqrt{\frac{2C_0}{n}}.$$

Proof. We start by observing from the definition of $\ell_s(\pi)$ that $\ell_i(\pi) = \sum_{t=1}^{H} \mathbb{E}_{s_t \sim d_t^{\pi^i}}[\ell_{s_t}(\pi)]$. Let us consider the loss difference

$$\ell_s(\pi) - \ell_s(\pi^\star) = -\sum_{a\in\mathcal{A}} \pi^\star(a \mid s)\log \pi(a \mid s) + \sum_{a\in\mathcal{A}} \pi^\star(a \mid s)\log \pi^\star(a \mid s) = D_{\mathrm{KL}}\big(\pi^\star(\cdot \mid s) \,\|\, \pi(\cdot \mid s)\big).$$

Thus,

$$\ell_i(\pi) - \ell_i(\pi^\star) = \sum_{t=1}^{H} \mathbb{E}_{s_t\sim d_t^{\pi^i}}\Big[D_{\mathrm{KL}}\big(\pi^\star(\cdot \mid s_t) \,\|\, \pi(\cdot \mid s_t)\big)\Big]. \tag{7}$$

We use shorthand $\pi^i(s) := \pi^i(\cdot \mid s)$ and $D_i(s) := D_{\mathrm{KL}}\big(\pi^\star(\cdot \mid s) \,\|\, \pi^i(\cdot \mid s)\big)$. Let us now consider the mirror map to be the negative entropy, so that the associated Bregman divergence is the KL divergence. We run the state-conditioned mirror descent update:

$$\pi^{i+1}(s) \in \arg\min_{q\in\Delta(\mathcal{A})} \left\{\langle \nabla \ell_s(\pi^i), q\rangle + \frac{1}{\eta} D_{\mathrm{KL}}(q\|\pi^i(s))\right\}.$$

From Lemma A.3, we have

$$\langle \eta\nabla\ell_s(\pi^i) + \nabla\Phi(\pi^{i+1}(s)) - \nabla\Phi(\pi^i(s)),\, \pi^\star(s) - \pi^{i+1}(s)\rangle \ge 0.$$

Upon rearranging, we have

$$\eta\langle \nabla\ell_s(\pi^i), \pi^{i+1}(s) - \pi^\star(s)\rangle \le \langle \nabla\Phi(\pi^{i+1}(s)) - \nabla\Phi(\pi^i(s)),\, \pi^\star(s) - \pi^{i+1}(s)\rangle.$$

Next, we use Lemma A.4 to obtain

$$\eta\langle \nabla\ell_s(\pi^i), \pi^{i+1}(s) - \pi^\star(s)\rangle \le D_i(s) - D_{i+1}(s) - D_{\mathrm{KL}}(\pi^{i+1}(s)\|\pi^i(s)). \tag{8}$$

Now, from assumption, we have,

$$\left|\frac{\partial \ell_s(\pi^i)}{\partial \pi(a \mid s)}\right| = \left|\frac{\pi^\star(a|s)}{\pi^i(a|s)}\right| \le C.$$

Next, we define

$$\Delta^i(s) := \pi^i(s) - \pi^{i+1}(s).$$

Then

$$\eta\langle \nabla\ell_s(\pi^i), \pi^i - \pi^{i+1}\rangle \le \eta C\|\Delta^i(s)\|_1.$$

Upon applying Young's inequality ($ab \le \frac{a^2}{2} + \frac{b^2}{2}$), with $a = \eta C$ and $b = \Delta^i(s)$, we get

$$\eta\langle \nabla\ell_s(\pi^i), \pi^i - \pi^{i+1}\rangle \le \frac{\eta^2 C^2}{2} + \frac{1}{2}\|\Delta^i(s)\|_1^2.$$

Using Pinsker's inequality,

$$\frac{1}{2}\|\Delta^i(s)\|_1^2 \le D_{\mathrm{KL}}(\pi^{i+1}(s)\|\pi^i(s)).$$

Thus,

$$\eta\langle \nabla\ell_s(\pi^i), \pi^i(s) - \pi^{i+1}(s)\rangle \le \frac{\eta^2 C^2}{2} + D_{\mathrm{KL}}(\pi^{i+1}(s)\|\pi^i(s)). \tag{9}$$

Now, from convexity of ℓ_s, we have,

$$\eta(\ell_s(\pi^i) - \ell_s(\pi^\star)) \le \eta\langle \nabla\ell_s(\pi^i),\, \pi^i(s) - \pi^{i+1}(s)\rangle + \eta\langle \nabla\ell_s(\pi^i),\, \pi^{i+1}(s) - \pi^\star(s)\rangle.$$

Upon combining Equations 8 and 9, we obtain

$$\eta(\ell_s(\pi^i) - \ell_s(\pi^\star)) \leq \frac{\eta^2 C^2}{2} + D_i(s) - D_{i+1}(s).$$

Now, we reuse the definition of $D_i(s) = \ell_s(\pi^i) - \ell_s(\pi^\star)$, this gives

$$\eta D_i(s) \leq \frac{\eta^2 C^2}{2} + D_i(s) - D_{i+1}(s).$$

Thus,

$$D_{i+1}(s) \leq (1-\eta) D_i(s) + \frac{\eta^2 C^2}{2}. \tag{10}$$

Next, we simplify this recursion. In particular, for a given state s, let us define

$$a_i := D_i(s), \quad b := \frac{\eta^2 C^2}{2}.$$

Therefore, Equation 10 can be re-written as

$$a_{i+1} \leq (1-\eta) a_i + b.$$

Using Lemma A.5, we get

$$D_i(s) \leq (1-\eta)^{i-1} D_1(s) + \frac{\eta^2 C^2}{2} \sum_{k=0}^{i-2} (1-\eta)^k, \quad \forall i \geq 1.$$

Next, we bound the geometric sum:

$$\sum_{k=0}^{i-2} (1-\eta)^k \leq \sum_{k=0}^{\infty} (1-\eta)^k = \frac{1}{\eta}.$$

Therefore,

$$D_i(s) \leq (1-\eta)^{i-1} D_1(s) + \frac{\eta^2 C^2}{2} \cdot \frac{1}{\eta} = (1-\eta)^{i-1} D_1(s) + \frac{\eta C^2}{2}.$$

Using $D_1(s) \leq C_0$, we obtain

$$D_i(s) \leq (1-\eta)^{i-1} C_0 + \frac{\eta C^2}{2}.$$

Now, considering Eq. 7, we have

$$\ell_i(\pi^i) - \ell_i(\pi^\star) = \sum_{t=1}^{H} \mathbb{E}_{s_t \sim d_t^{\pi^i}} [D_i(s_t)] \leq H(1-\eta)^{i-1} C_0 + H \frac{\eta C^2}{2}.$$

Upon summing from $i = 1$ to n gives

$$\begin{aligned}
\sum_{i=1}^{n} (\ell_i(\pi^i) - \ell_i(\pi^\star)) &\leq H \sum_{i=1}^{n} \left((1-\eta)^{i-1} C_0 + \frac{\eta C^2}{2} \right) \\
&= \frac{n \eta H C^2}{2} + H C_0 \sum_{i=1}^{n} (1-\eta)^{i-1} \\
&\leq \frac{n \eta H C^2}{2} + \frac{H C_0}{\eta} \qquad \left(\because \sum_{i=1}^{n} (1-\eta)^{i-1} \leq \frac{1}{\eta} \right)
\end{aligned}$$

Upon dividing by n gives

$$\frac{\sum_{i=1}^{n} (\ell_i(\pi^i) - \ell_i(\pi^\star))}{n} \leq \frac{\eta H C^2}{2} + \frac{H C_0}{\eta n}$$

Finally, choosing $\eta = \sqrt{\frac{2C_0}{C^2 n}}$ completes the proof. □

We now introduce the quantities that characterize the difficulty of imitation under a stochastic teacher, following Foster et al. (2024).

Teacher variance and recoverability. Following Foster et al. (2024), for any policy π, define

$$\bar{\sigma}^2_{\pi_T|\pi} := \sum_{h=1}^{H} \mathbb{E}_{P^{\pi \circ_h \pi_T}} \left[\left(Q_h^{\pi_T}(x_h, a_h) - V_h^{\pi_T}(x_h) \right)^2 \right],$$

where $P^{\pi \circ_h \pi_T}$ denotes the trajectory distribution that follows π up to time $h-1$ and switches to π_T thereafter. We then define the worst-case teacher variance

$$\bar{\sigma}^2_{\pi_T} := \sup_{\pi} \bar{\sigma}^2_{\pi_T|\pi}.$$

We also define the signed recoverability parameter

$$\mu_T := \max_{x \in \mathcal{X},\, a \in \mathcal{A},\, h \in [H]} \left| Q_h^{\pi_T}(x, \pi_{T,h}(x)) - Q_h^{\pi_T}(x, a) \right|.$$

Remarks. The quantity $\bar{\sigma}^2_{\pi_T}$ captures the stochasticity of the teacher policy, and in particular, $\bar{\sigma}^2_{\pi_T} = 0$ when π_T is deterministic. The constant μ_T measures the worst-case degradation in value from taking a suboptimal action under the teacher's value function, and satisfies $0 \le \mu_T \le R$ when rewards are bounded by R. Together, these quantities characterize the difficulty of imitation. When μ_T is small, the environment is forgiving and errors are easily recoverable; in contrast, large μ_T corresponds to settings where early mistakes can lead to significant long-term loss. Moreover, the two quantities are related via the bound $\bar{\sigma}^2_{\pi_T} \le \mu_T^2 H$, which implies that the stochasticity of the teacher is controlled by the recoverability of the environment.

Theorem 1. Let π_S be the student policy returned by the updates in Eq. (4). DistIL ensures that

$$J(\pi_T) - J(\pi_S) \lesssim \sqrt{\bar{\sigma}^2_{\pi_T} \frac{HC}{2}} \left(\frac{2C_0}{n} \right)^{1/4} + \frac{\mu_T H C}{2} \sqrt{\frac{2C_0}{n}}.$$

Proof. Let us start by proving the first case. For a fixed state s_t, we define the statewise cross-entropy

$$\ell_{s_t}(\pi) := -\sum_{a \in \mathcal{A}} \pi_T(a \mid s_t) \log \pi(a \mid s_t).$$

Then, $\ell_{s_t}(\pi) - \ell_{s_t}(\pi_T) = D_{\mathrm{KL}}\big(\pi_T(\cdot \mid s_t) \,\|\, \pi(\cdot \mid s_t)\big)$. Therefore, using the definition of ℓ_i,

$$\ell_i(\pi^i) - \ell_i(\pi_T) = \sum_{t=1}^{H} \mathbb{E}_{s_t \sim d_t^{\pi^i}} \left[D_{\mathrm{KL}}\big(\pi_T(\cdot \mid s_t) \,\|\, \pi^i(\cdot \mid s_t)\big) \right].$$

Averaging over i gives

$$\epsilon_n := \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{H} \mathbb{E}_{s_t \sim d_t^{\pi^i}} \left[D_{\mathrm{KL}}\big(\pi_T(\cdot \mid s_t) \,\|\, \pi^i(\cdot \mid s_t)\big) \right].$$

Now, for any pair of distributions (p, q), we have $D^2_{\mathrm{H}}(p, q) \le \frac{1}{2} D_{\mathrm{KL}}(p \| q)$, where D_{H} is the Hellinger distance. Applying this pointwise with $p = \pi_T(\cdot \mid s_t)$ and $q = \pi^i(\cdot \mid s_t)$, then taking expectation over $s_t \sim d_t^{\pi^i}$, summing over t, and averaging over i, gives

$$\mathcal{E}_n := \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{H} \mathbb{E}_{s_t \sim d_t^{\pi^i}} \left[D^2_{\mathrm{H}}\big(\pi_T(\cdot \mid s_t), \pi^i(\cdot \mid s_t)\big) \right] \le \frac{1}{2} \epsilon_n.$$

Next, we make use of the Proposition C.1 of (Foster et al., 2024), which works for our online setting, to bound the rollout suboptimality by a variance term and a recoverability term as

$$J(\pi_T) - J(\pi_S) \lesssim \sqrt{\bar{\sigma}^2_{\pi_T} \mathcal{E}_n} + \mu_e \mathcal{E}_n.$$

Substituting $\mathcal{E}_n \le \epsilon_n/2$ gives

$$J(\pi_T) - J(\pi_b) \;\lesssim\; \sqrt{\frac{\bar{\sigma}^2_{\pi_T}}{2}\,\epsilon_n} + \frac{\mu_T}{2}\,\epsilon_n.$$

Finally, plugging in the Theorem B.4 gives

$$J(\pi_T) - J(\pi_b) \;\lesssim\; \sqrt{\bar{\sigma}^2_{\pi_T}\frac{HC}{2}}\left(\frac{2C_0}{n}\right)^{1/4} + \frac{\mu_T HC}{2}\sqrt{\frac{2C_0}{n}}.$$

□

B.8. Connections with Max RL

We show that the forward cross-entropy objective induces a lower bound on a teacher-weighted maximum likelihood RL objective.

Proposition 6. For a fixed prompt x, we define the success probability $p_\pi(x) := \mathbb{E}_{y\sim\pi(\cdot|x)}[r(x,y)]$, where $r(x,y) \in \{0,1\}$ indicates correctness. Then

$$-H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) \le p_{\pi_T}(x)\log p_{\pi_\theta}(x),$$

where

$$p_{\pi_\theta}(x) := \mathbb{P}_{y\sim\pi_\theta(\cdot|x)}[r(x,y)=1].$$

Proof. To show this, we begin with the definition of forward cross-entropy:

$$H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) = \sum_y \pi_T(y \mid x)\big[-\log \pi_\theta(y \mid x)\big].$$

Next, we partition the sum into correct and incorrect responses:

$$H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) = \sum_{y:r(x,y)=1} \pi_T(y \mid x)\big[-\log \pi_\theta(y \mid x)\big] + \sum_{y:r(x,y)=0} \pi_T(y \mid x)\big[-\log \pi_\theta(y \mid x)\big].$$

For any y such that $r(x,y)=1$, we can write

$$\pi_\theta(y \mid x) = p_{\pi_\theta}(x)\,\pi_\theta(y \mid x, r=1),$$

where

$$p_{\pi_\theta}(x) = \mathbb{P}_{y\sim\pi_\theta(\cdot|x)}[r(x,y)=1].$$

Thus,

$$-\log \pi_\theta(y \mid x) = -\log p_{\pi_\theta}(x) - \log \pi_\theta(y \mid x, r=1).$$

Substituting this into the first term, we get

$$\begin{aligned} H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) = -p_{\pi_T}(x)\log p_{\pi_\theta}(x) + p_{\pi_T}(x)\,H^{\times}\big(\pi_T(\cdot \mid x, r=1), \pi_\theta(\cdot \mid x, r=1)\big) \\ + (1-p_{\pi_T}(x))\,\mathbb{E}_{y\sim\pi_T(\cdot|x,r=0)}[-\log \pi_\theta(y \mid x)]. \end{aligned}$$

Next, we define

$$\mathcal{R}(x) := p_{\pi_T}(x)\,H^{\times}\big(\pi_T(\cdot \mid x, r=1), \pi_\theta(\cdot \mid x, r=1)\big) + (1-p_{\pi_T}(x))\,\mathbb{E}_{y\sim\pi_T(\cdot|x,r=0)}[-\log \pi_\theta(y \mid x)].$$

Since both terms are nonnegative, we have $\mathcal{R}(x) \ge 0$. Therefore,

$$H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) = -p_{\pi_T}(x)\log p_{\pi_\theta}(x) + \mathcal{R}(x) \ge -p_{\pi_T}(x)\log p_{\pi_\theta}(x).$$

Rearranging gives

$$-H^{\times}\big(\pi_T(\cdot \mid x), \pi_\theta(\cdot \mid x)\big) \le p_{\pi_T}(x)\log p_{\pi_\theta}(x).$$

□

C. Extended Related Work

Credit Assignment in RLVR. RLVR enables training from automatically verifiable outcomes without human preference annotations (Shao et al., 2024; Guo et al., 2025). However, supervision is applied at the trajectory level, assigning identical signals to all tokens despite their differing roles, making credit assignment especially challenging for long-horizon reasoning. One line of work addresses this by introducing intermediate supervision via process-reward models or step-wise evaluators (Lightman et al., 2023; Zhang et al., 2025; Wang et al., 2024; Yang et al., 2026b; Luo et al., 2024; Chen et al., 2024; Zhang et al., 2024; Yang et al., 2025b; Dai et al., 2025; Cui et al., 2025). While effective, these approaches require additional annotation or auxiliary models. An alternative direction avoids explicit supervision by reweighting updates using internal signals such as entropy or uncertainty, but these rely on heuristic notions of token importance (Xie et al., 2025; Li et al., 2026c; Cheng et al., 2026; Wang et al., 2025; Chen et al., 2025a; Sun et al., 2025; Li et al., 2025; Chen et al., 2025b). In our work, instead of estimating token importance heuristically or learning separate reward models, we construct token-level supervision directly through privileged feedback. This yields structured credit assignment while preserving the simplicity of the verifier-only training pipeline.

On-Policy Distillation and On-Policy Self-Distillation. On-policy distillation trains a student on its own rollouts while a teacher provides token-level supervision via divergence-based objectives (Agarwal et al., 2024; Xu et al., 2024; Lu & Lab, 2025; Xiao et al., 2026; Yang et al., 2025a). Aligning training with the student's visitation distribution reduces the distribution mismatch inherent in off-policy approaches, but it typically relies on a stronger external teacher. Recent work extends this paradigm to self-distillation, where a single model provides both supervision and updates. OPSD (Zhao et al., 2026) formulates this as minimizing the divergence between a privileged teacher, which is conditioned on ground-truth reasoning traces, and a student, thereby improving token efficiency. SDPO (Hübotter et al., 2026) generalizes this setup to incorporate arbitrary feedback signals (e.g., code execution error logs), while RLSD (Yang et al., 2026a) integrates self-distillation with the GRPO methodologically to stabilize training. However, self-distillation introduces new challenges. It can degrade reasoning by suppressing epistemic uncertainty (Kim et al., 2026), and its success depends critically on alignment between teacher and student distributions as well as the quality of self-generated targets (Li et al., 2026b). Recent methods address these issues through improved supervision and routing strategies, including competence-aware weighting (PACED), iterative self-revision (SD-Zero), and hybrid routing between distillation and RL updates (SRPO) (Xu et al., 2026; He et al., 2026; Li et al., 2026a). In contrast, our approach revisits the objective underlying self-distillation. We adopt a cross-entropy-based formulation with well-behaved optimization properties, resulting in more stable training dynamics and improved gradient estimation. This is related to methods such as DAgger (Ross et al., 2011) in imitation learning that query the expert for supervision on the learner's induced state distribution.

D. Experiment Details

We use 4 H200 GPUs for all our experiments. For prompts and implementation, we use SDPO's codebase (LAS Group, ETH Zurich, 2026) on Science and Coding benchmarks whereas OPSD's codebase (Zhao, 2026) on Mathematics benchmarks.

For DistIL, we observe performance degradation for longer sequences $y \sim \pi_\theta(\cdot \mid x)$, as the cumulative future loss $\sum_{i>t} \mathrm{H}^{\times}(\cdot)$ grows with sequence length, leading to disproportionately large gradient magnitudes. To mitigate this effect, we apply length normalization, similar to Gu et al. (2024). Specifically, we replace the cumulative loss in future-credit assignment in Eq. (5) with

$$\tilde{L}_{t+1} = \frac{1}{T-t-1} \sum_{i>t} \mathrm{CE}\big(\mathrm{stopgrad}\big(\pi_\theta(\cdot \mid x, y_{1:i-1}, f)\big) \parallel \pi_\theta(\cdot \mid s_i)\big).$$

D.1. Hyperparameters for Science and Coding Benchmark

For Science and Coding benchmarks, we follow the implementation of SDPO and use their reported hyperparameters for baselines. The hyperparameters for SDPO, DistIL (Ours), and GRPO with its off-policy and on-policy variants are provided in Table 3, 4, and 5 respectively.

D.2. Hyperparameters for Maths task

For Mathematics benchmarks, we follow the implementation of OPSD and use their reported hyperparameters for baselines. The hyperparameters for SDPO, OPSD, DistIL (Ours), and GRPO are provided in Table 6 and that for SFT in Table 7.

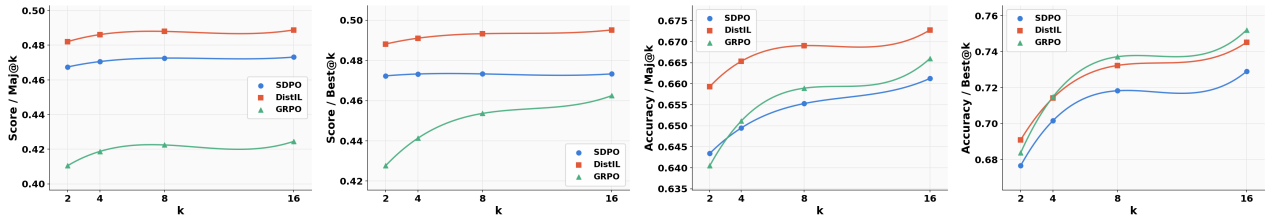

Figure 3. LCBv6 evaluation at τ=0.2 (checkpoint-step 80), reporting Score and Accuracy at Best@k and Maj@k for $k \in \{2, 4, 8, 16\}$.

Evaluation hyperparameters are provided in Table 8 for Qwen3-4B-2507-Instruct model and Table 9 for Qwen3-8B model. The only difference between OPSD and SDPO in this scenario is that OPSD uses Forward-KL divergence whereas SDPO uses reverse-KL divergence in these mathematics experiments.

E. Additional Results

E.1. More results for Coding.

SDPO evaluated their coding experiment at temperature = 0.6. We include results for that scenario in here. DistIL achieves Acc/Mean@16 $= 0.660$ and Score/Mean@16 $= 0.482$, outperforming SDPO $(0.643, 0.467)$ and GRPO $(0.639, 0.411)$. The large gap over GRPO reflects its fundamental inability to exploit execution feedback. Figure 3 notice that the same trend follows as that for temperature=0.2, except that GRPO becomes slightly better on Accuracy/Best@k for k=16.

E.2. Ablation Study

E.2.1. Credit Assignment

We compare DistIL against a CE (cross-entropy) baseline that retains only the cross-entropy term of the gradient, performing local credit assignment analogous to SDPO and OPSD. Figure 4 shows results on the Material domain. DistIL consistently outperforms CE throughout training, demonstrating that the full credit assignment across the entire response distribution rather than assigning it locally, is the key driver of performance gains. The CE baseline exhibits higher variance and fails to improve to the level of DistIL's performance, suggesting that local credit assignment is insufficient for learning robust scientific reasoning.

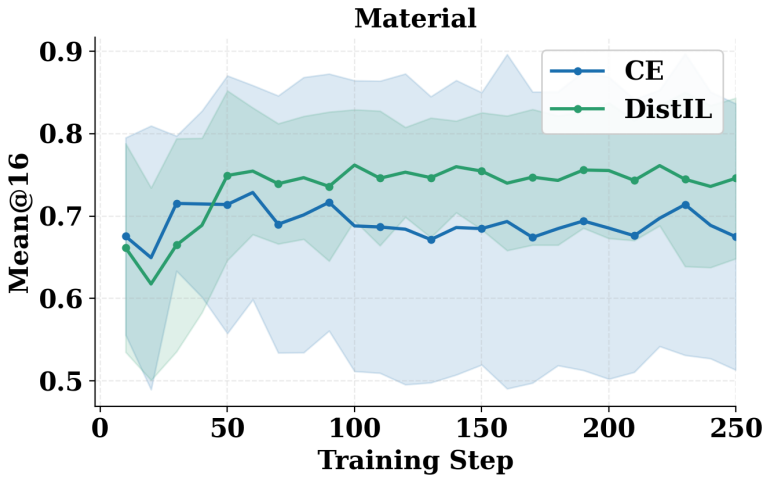

Figure 4. Comparison between DistIL (Ours) and CE. The difference between these methods lies in the credit assignment. CE only does local token-wise creadit assignment similar to SDPO and OPSD, whereas DistIL does full credit assignment as given in Eq. 5.

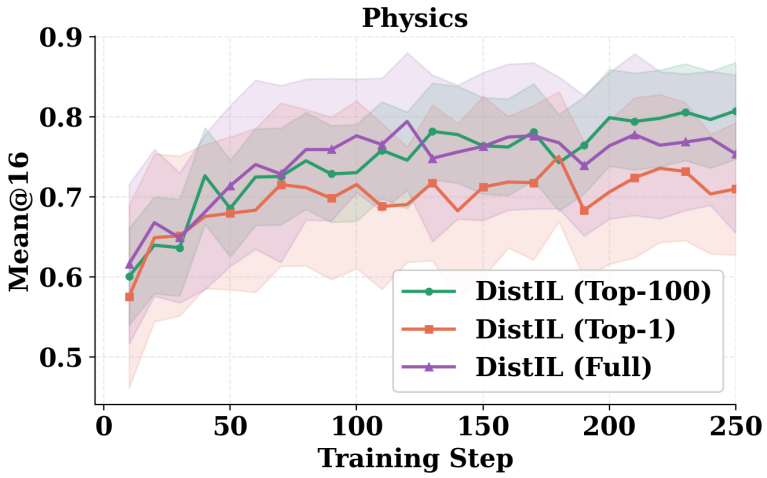

Figure 5. Comparison of Avg@16 between different number of top-k tokens used for training DistIL on Material benchmark.

E.2.2. Top-1 v/s Top k v/s Full distribution

We ablate the number of teacher tokens used in the distillation objective across $K \in \{1, 100, \text{Full}\}$ on the Physics domain (Figure 5). Top-1 distillation, which aligns the student only to the single highest-probability teacher token, underperforms throughout training, reflecting insufficient signal from a single token. Top-100 achieves the best final performance and most stable training trajectory, outperforming both Top-1 and Full distillation. Full distillation, while competitive, introduces noise from low-probability tokens that are unlikely to be informative, slightly degrading performance relative to Top-100. Based on this analysis, we use K=100 in all experiments.

Parameters	**Science**	**Coding**
General		
Model	Qwen/Qwen3-8B allenai/Olmo3-7B-Instruct	Qwen/Qwen3-8B
Thinking	False	False
Data		
Max. prompt length	2048	2048
Max. response length	8192	8192
Batching		
Question batch size	32	32
Mini batch size	32	1
Number of rollouts	8	8
Rollout		
Inference engine	vllm	vllm
Temperature	1.0	1.0
Validation		
Number of rollouts	16	4
Temperature	0.6	0.6
Top-p	0.95	0.95
SDPO Loss		
Top-K distillation	100	20
Distillation divergence	Jensen–Shannon	Reverse-KL
Clip advantages	–	–
Teacher-EMA update rate	0.05	0.01
Rollout importance sampling clip	2	2
Training		
Optimizer	AdamW	AdamW
Learning rate	1×10^{-5} (constant)	1×10^{-6} (constant)
Warmup steps	10	0
Weight decay	0.01	0.01
Gradient clip norm	1.0	1.0

Table 3. Hyperparameters used for **SDPO** for each experimental setup.

Parameters	**Science**	**Coding**
General		
Model	Qwen/Qwen3-8B allenai/Olmo3-7B-Instruct	Qwen/Qwen3-8B
Thinking	False	False
Data		
Max. prompt length	2048	2048
Max. response length	8192	8192
Batching		
Question batch size	32	32
Mini batch size	32	1
Number of rollouts	8	8
Rollout		
Inference engine	vllm	vllm
Temperature	1.0	1.0
Validation		
Number of rollouts	16	4
Temperature	0.6	0.6
Top-p	0.95	0.95
SDPO Loss		
Top-K distillation	100	20
Distillation divergence	Jensen–Shannon	Reverse-KL
Clip advantages	–	–
Teacher-EMA update rate	0.01	0.01
Rollout importance sampling clip	2	2
Training		
Optimizer	AdamW	AdamW
Learning rate	5×10^{-5} (constant)	1×10^{-6} (constant)
Warmup steps	10	0
Weight decay	0.01	0.01
Gradient clip norm	1.0	1.0

Table 4. Hyperparameters used for **DistIL** for each experimental setup.

Parameters	Value
General	
Model	Qwen/Qwen3-8B allenai/Olmo3-7B-Instruct
Thinking	False
Data	
Max. prompt length	2048
Max. response length	8192
Batching	
Question batch size	32
Mini batch size	8 (default) / 32 (on-policy)
Number of rollouts	8
Rollout	
Inference engine	vllm
Temperature	1.0
Validation	
Temperature	0.6
Top-p	0.95
Number of rollouts	16
Loss	
ϵ-high	0.28
Rollout importance sampling clip	2
KL coefficient (λ)	0.0
Training	
Optimizer	AdamW
Learning rate	1×10^{-6} (default) / 1×10^{-5} (on-policy)
Warmup steps	10
Weight decay	0.01
Gradient clip norm	1.0

Table 5. Hyperparameters used for **GRPO** (On-Policy) and (Off-Policy) for Science and Coding.

Parameter	GRPO	OPSD/SDPO/DistIL
Learning Rate	5×10^{-6}	5×10^{-6}
Effective Batch Size	32	32
LoRA Rank (r)	64	64
LoRA Alpha (α)	128	128
LoRA Target Modules	`q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj`	
Max Completion Length	16,000	16,384
Number of Generations per Prompt	8	1
Sampling Temperature	0.7	0.7
KL Coefficient (β)	0.0	–
Training Steps	500	100

Table 6. Training configuration for GRPO, OPSD, SDPO, DistIL on Mathematics benchmarks.

Parameter	SFT
Learning Rate	5×10^{-6}
Effective Batch Size	32
LoRA Rank (r)	64
LoRA Alpha (α)	128
LoRA Target Modules	`q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj`
Max Sequence Length	16,000
Number of Training Steps	100

Table 7. Training configuration for SFT on Mathematics benchmarks.

Parameter	**Value**
Max New Tokens	16,384
Thinking Mode	Enabled
Top-p	0.95
Top-k	20
Min-p	0.0
Presence Penalty	0.0
Samples per Prompt	64
Temperature	0.7

Table 8. Evaluation parameters for Qwen3-4B-Instruct-2507 Model on Maths Task.

Parameter	**Value**
Max New Tokens	38,912
Thinking Mode	Enabled
Top-p	1.0
Top-k	-1
Min-p	0.0
Presence Penalty	0.0
Samples per Prompt	64
Temperature	1.0

Table 9. Evaluation parameters for Qwen3-8B Model on Maths Task.