A DISTRIBUTION-MATCHING PERSPECTIVE ON LLM TRAINING ALGORITHMS

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we propose a unified framework for training algorithms across different stages of modern large language model development, based on the perspective of distribution matching. We show that a broad class of language model training algorithms can be interpreted as minimizing a certain divergence between the current model and a target distribution, and that characterizing the properties of this target distribution provides substantial insight into the behavior of the corresponding training algorithms. By analyzing specific forms of the target distribution, we establish that On-Policy Distillation (OPD) enjoys an advantage over classical Knowledge Distillation (KD) when the current model distribution is close to the target distribution, while exhibiting a disadvantage when the two distributions are far apart. Furthermore, when the target distribution itself depends on the model being optimized, we show that the accumulation of estimation variance can induce a deterministic degradation of the target distribution. These results provide a theoretical explanation, at least in part, for the long-term behavior of existing Reinforcement Learning (RL) algorithms. With this intuition we further designed Conditional KL (CKL) regularization for mitigating the degradation of the target distribution. CKL can effectively preserve mode diversity while relaxing the constraint imposed by classical KL regularization on reward improvements. We believe this theoretical perspective can bring some new understanding of language model training algorithms and provide useful guidance for future research.

1 INTRODUCTION

Modern Large Language Models (LLMs) require a complex training pipeline with different algorithms from pre-training to post-training (Team et al., 2026; DeepSeek-AI et al., 2026; GLM-5-Team et al., 2026). These algorithms are motivated by different intuitions. For example, the objectives of pre-training and Supervised Fine-Tuning (SFT) arise directly from distribution matching under the cross-entropy loss; On Policy Distillation (OPD) is constructed around on-policy sampling (Lochab et al., 2026); Reinforcement Learning (RL) aims to maximize the expected reward (Williams, 1992); and MaxRL can be viewed as arising from maximum-likelihood estimation of correct solutions (Tajwar et al., 2026). We hope to identify the commonalities and essential differences among these algorithms, which will provide a more unified understanding and insight into the study of their properties.

Many of the training algorithms of LLMs already admit a distribution matching interpretation (Korbak et al., 2022a; Khalifa et al., 2021), whereas RL is often viewed fundamentally distinct. In this work we first studied the property of an ideal update step in RL and show that, under binary rewards, it preserves the conditional distribution associated with a given reward value. Moreover, if we consider the forward KL divergence between the current model and the true conditional distribution, we will obtain the training objective of MaxRL (Tajwar et al., 2026), which is closely related to RL. Similar results can also be extended to RL with nonbinary rewards.

This provides a unified perspective that places the mainstream training algorithms for LLMs within a common distribution-matching framework. Analyzing how the target distribution changes and affects the learning dynamics under different conditions can yield a variety of interesting theoretical results that characterize the properties of these algorithms.

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

Depending on whether the target distribution being optimized depends on the current model distribution, these algorithms can be further categorized into iterative and non-iterative methods. In what follows, we progressively examine the properties of three classes of algorithms, from simpler to more complex cases: non-iterative methods, iterative methods with zero-mean gradients, and iterative methods with nonzero-mean gradients.

For the first class, we mainly study OPD and SFT-based knowledge distillation (Hinton et al., 2015) algorithms. By analyzing the variance induced by different target distributions under different assumptions, we roughly classified the boundary of effectiveness of these algorithms, providing a theoretical explanation for previous empirical observations regarding the stability of distillation algorithms (Li et al., 2026; Oh et al., 2026).

The second class primarily consists of the iterative self-distillation algorithms. These methods lie between distillation and RL, while retaining theoretical properties that are more amenable to analysis. Within this class, we identify an intriguing *variance accumulation* phenomenon in iterative training, which leads to a monotonic decrease in the expected entropy of the target distribution. In contrast to previous local results derived from training dynamics (Cui et al., 2025; Ren & Sutherland, 2025; Jin et al., 2026), our analysis further shows that, although the model entropy may either increase or decrease after an individual update, its long-term behavior exhibits a systematic downward trend under the variance accumulation.

The third class mainly concerns of RL algorithms. In realistic training settings, however, the target distribution invariance of RL and MaxRL under idealized conditions is no longer preserved. Their target distributions instead exhibit collapse, driven in part by effects such as variance accumulation. Such target distribution drift serves as an important manifestation of the discrepancy between practical training dynamics and their idealized counterparts. inspired by the analysis on the target distribution, we explored Conditional KL (CKL) regularization in RL. CKL regularization directly controls the drift of the target distribution in RL, preserving the diversity of correct samples with relaxed restriction on reward gains.

2 A Unified Framework of LLM Training Algorithms

2.1 Pre-Training And Distillation Algorithms

In pre-training and Supervised Fine-Tuning (SFT), minimizing the standard cross-entropy loss is equivalent to minimizing the forward KL divergence from the data distribution to the model distribution. To make this precise, let $\mathcal{D} = \{x^{(i)}\}_{i=1}^{N}$ be a fixed, nonempty dataset of token sequences, let $p_{\mathcal{D}}(x) = N^{-1}\sum_{i=1}^{N} \mathbf{1}\{x^{(i)} = x\}$ be its empirical distribution, and let $\pi_\theta(x)$ be the model probability of sequence x with parameter θ. Then

$$\begin{aligned} D_{\mathrm{KL}}(p_{\mathcal{D}} \| \pi_\theta) &= \mathbb{E}_{x\sim p_{\mathcal{D}}}\left[\log \frac{p_{\mathcal{D}}(x)}{\pi_\theta(x)}\right] = -\mathbb{E}_{x\sim p_{\mathcal{D}}}[\log \pi_\theta(x)] - H(p_{\mathcal{D}}) \\ &= \mathcal{L}_{\mathrm{CE}}(\theta) - H(p_{\mathcal{D}}), \end{aligned} \tag{1}$$

Here the data entropy $H(p_{\mathcal{D}}) = -\mathbb{E}_{x\sim p_{\mathcal{D}}}[\log p_{\mathcal{D}}(x)]$ is finite and independent of θ which has no effect on the gradient. This proves the equivalence.

For SFT, let x denote a prompt and y its target response, with fixed empirical joint distribution $p_{\mathcal{D}}(x, y)$. The same decomposition holds conditionally:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim p_{\mathcal{D}}}[\log \pi_\theta(y \mid x)] \tag{2}$$

$$= \mathbb{E}_{x\sim p_{\mathcal{D}}}\left[D_{\mathrm{KL}}\big(p_{\mathcal{D}}(\cdot \mid x) \| \pi_\theta(\cdot \mid x)\big)\right] + H_{p_{\mathcal{D}}}(Y \mid X), \tag{3}$$

Under the autoregressive factorization, $-\log \pi_\theta(y \mid c) = -\sum_{t=1}^{|y|} \log \pi_\theta(y_t \mid c, y_{<t})$, this sequence-level objective is exactly the sum of token-level cross-entropies with ground-truth prefixes.

In many applications the response could come from a teacher model π_t, instead of the dataset. Then the training objective of SFT would become minimizing the forward KL divergence between the

teacher model and the student model. This yields the classical SFT-based Knowledge-Distillation (KD) algorithm (Hinton et al., 2015), which is often simply called SFT as well:

$$\mathcal{J}_{\mathrm{KD}} = -\mathbb{E}_{x\sim\mathcal{D}}[D_{\mathrm{KL}}\big(\pi_t(\cdot \mid x)\|\pi_\theta(\cdot \mid x)\big)] = \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_t(\cdot|x)}[\log \pi_\theta(y \mid x)] - H(\pi_t). \tag{4}$$

OPD recently emerged as a new algorithm for distillation tasks (Thinking Machines Lab & Lu, 2025). It can exhibit superior performance than SFT in some cases, and received great attention from the industry (Team et al., 2026; DeepSeek-AI et al., 2026; GLM-5-Team et al., 2026). While in some cases this algorithm may also be unstable and failed to complete the distillation task (Li et al., 2026). The key difference between OPD and KD is that, OPD uses sequence-level reverse KL divergence as the training objective:

$$\mathcal{J}_{\mathrm{OPD}} = -\mathbb{E}_{x\sim\mathcal{D}}[D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid x)\|\pi_t(\cdot \mid x)\big)] = \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(\cdot|x)}[\log \frac{\pi_t(y \mid x)}{\pi_\theta(y \mid x)}]. \tag{5}$$

We should note that the sequence-level KL objective directly yields sampled-token OPD. While there exist several variants of this OPD (Li et al., 2026), this sampled-token OPD is indeed adopted by the industry (Team et al., 2026) and share some important properties with other variants. the following analysis will primarily be built upon this algorithm.

2.2 Reinforce Learning Algorithms

The objective of RL is to maximize the expected reward r given to each possible action of the policy model, while for LLMs an action is typically a sequence generation. Its standard objective and gradient are (Williams, 1992):

$$\mathcal{J}_{\mathrm{RL}} = \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(\cdot|x)}[r(x,y)], \qquad \nabla_\theta\mathcal{J}_{\mathrm{RL}} = \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(\cdot|x)}[r(x,y)\nabla_\theta \log \pi_\theta]. \tag{6}$$

Modern reinforcement learning algorithms such as PPO (Schulman et al., 2017b) and GRPO (Shao et al., 2024) additionally introduce various methods, including advantage estimation and importance sampling techniques, to better achieve stable training, but they are essentially optimizing the same training objective.

In fact, this objective also implicitly introduces a target distribution. We first characterize an ideal policy update that improves expected reward while remaining close to the current policy under a forward-KL restriction. (Similar result also holds for reverse-KL restriction)

theorem 2.1 (Reward reweighting under a forward-KL restriction)**.** *For a prompt x, the ideal update maximizing $\mathbb{E}_{y\sim q}[r(x,y)] - \beta D_{\mathrm{KL}}(\pi_\theta\|q)$ over response distributions q, with KL penalty $\beta > 0$, is*

$$q^*(y \mid x) = \frac{\beta\pi_\theta(y \mid x)}{\lambda_x - r(x,y)}, \tag{7}$$

where λ_x normalizes the distribution. Responses with the same reward receive the same probability multiplier, preserving their conditional distribution.

Proof: Appendix C.

Under binary rewards, we can verify that this update preserves the true conditional distribution $\pi_\theta^+(\cdot \mid x) = \pi_\theta(\cdot \mid x, r(x,y) = 1)$. If we use this distribution as the training target, matching this conditional distribution with the forward KL gives the following identity.

theorem 2.2 (True conditional KL and the MaxRL). *Let $\rho_\theta(x) = \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[r(x,y)]$ denote the success probability. Matching the true conditional target π_θ^+ by minimizing its forward KL to the current policy yields*

$$-\mathbb{E}_{x\sim\mathcal{D}} D_{\mathrm{KL}}\big(\pi_\theta^+(\cdot \mid x)\|\pi_\theta(\cdot \mid x)\big) = \mathbb{E}_{x\sim\mathcal{D}}[\log \rho_\theta(x)] = \mathcal{J}^{(\infty)}_{\mathrm{MaxRL}}(\theta). \tag{8}$$

Proof: Appendix D.

This actually gives the limit objective function of MaxRL algorithm, which is previously derived from the maximum likelihood estimator of the predictive accuracy (Tajwar et al., 2026). It is believed to have some superior performance on maintaining the diversity of the distribution. The standard RL objective is the first-order approximation of MaxRL. They induce the same gradient direction for an individual sample, differing only in the average weighting assigned across samples. Since our subsequent analysis focuses on individual samples, the two methods are effectively equivalent in the properties relevant to our analysis.

An auxiliary binary acceptance variable extends the conditional-KL identity to bounded nonbinary rewards on an augmented response space; Appendix D.1 outlines this construction, while our main RL analysis remains focused on binary rewards.

Table 1: Training algorithms viewed as distribution matching. Parameter-independent constants are omitted. Conditional objectives are averaged over prompts $x \sim \mathcal{D}$. Forward KL places the target in the first argument; Reverse KL places it in the second. RL and MaxRL use binary rewards and $\rho_\theta(x) > 0$, with the policy-dependent target π_θ^+. Algorithms are iterative if their target distribution depends on the current model, and noniterative otherwise; π_θ is highlighted in iterative targets. † RL is its first-order approximation.

Algorithm	Objective function	Target distribution	Divergence
Noniterative algorithms			
Pre-training	$\mathbb{E}_{x\sim p_\mathcal{D}}[\log \pi_\theta(x)]$	$p_\mathcal{D}(x)$	Forward KL
SFT	$\mathbb{E}_{y\sim p_\mathcal{D}(\cdot\|x)}[\log \pi_\theta(y \mid x)]$	$p_\mathcal{D}(y \mid x)$	Forward KL
SFT-based KD	$\mathbb{E}_{y\sim \pi_t(\cdot\|x)}[\log \pi_\theta(y \mid x)]$	$\pi_t(y \mid x)$	Forward KL
OPD	$\mathbb{E}_{y\sim\pi_\theta(\cdot\|x)}\left[\log \frac{\pi_t(y \mid x)}{\pi_\theta(y \mid x)}\right]$	$\pi_t(y \mid x)$	Reverse KL
Iterative algorithms			
RL	$\mathbb{E}_{y\sim\pi_\theta(\cdot\|x)}[r(x,y)]$	${\pi_\theta}^+ = \frac{r(x,y)\pi_\theta(y \mid x)}{\rho_\theta(x)}$	Forward KL†
MaxRL	$\log \mathbb{E}_{y\sim\pi_\theta(\cdot\|x)}[r(x,y)]$	${\pi_\theta}^+ = \frac{r(x,y)\pi_\theta(y \mid x)}{\rho_\theta(x)}$	Forward KL

3 The Effectiveness Border of Distillation Algorithms

When OPD algorithm was initially introduced, its superior performance was believed to come from its on policy nature (Thinking Machines Lab & Lu, 2025). SFT trains the student on the trajectory generated by the teacher. Even if the student learns well on prediction over teacher's prefix, if it can not generate a prefix with the same quality, the performance will still be limited. OPD method however, directly uses teacher's predictive distribution to correct the student on its own trajectory, and avoids this drawback (Thinking Machines Lab & Lu, 2025). However, this discussion ignored another important impact in OPD: The teacher's probability prediction on the student's prefix may not be sufficiently reliable as well. In some previous works we did observe some cases where OPD may face stability issue and failed to improve the student model (Li et al., 2026). Clearly this issue does not exist in SFT.

It is difficult to determine precisely which of the two issues discussed above has a dominant impact on performance. Beyond such relatively subjective analysis, however, we aim to establish rigorous

theoretical results and characterize the regimes in which the two algorithms are effective. Since all distillation methods have the same target distribution, one primary theoretical difference may lie in the estimation of the gradient. This leads to some interesting results.

3.1 Proximal Behavior

We study the proximal regime, where the student distribution is close to the teacher. At the *proximal limit*, they coincide, while the teacher is held fixed during differentiation. Both negative-KL population objectives and their gradients then vanish, but their sampled gradient estimators behave differently.

Lemma 3.1 (Gradient noise at the proximal limit)**.** *Using m independent responses to a prompt x. At $\pi_{\theta_0}(\cdot \mid x) = \pi_t(\cdot \mid x)$, both gradient estimators are unbiased with mean zero, but*

$$\widehat{g}_{\mathrm{OPD}}(\theta_0) = 0 \quad \textit{almost surely}, \qquad \mathrm{Cov}\big(\widehat{g}_{\mathrm{SFT}}(\theta_0)\big) = \frac{1}{m} F_x(\theta_0), \tag{9}$$

where $F_x(\theta_0)$ is the student Fisher information at prompt x. Thus SFT typically has nonzero gradient noise at, whereas OPD vanishes for every sampled batch.

Proof: Appendix E.

Under the smoothness assumptions, this strict gap still persists in a neighborhood of θ_0: OPD has lower mean-squared error in estimating its population gradient than SFT has. This establishes a local advantage in gradient estimation in a range of proximal case, and could provide a theoretical explanation on the superior performance of OPD.

3.2 Distal Behavior

The distal regime can exhibit different behavior. When the student assigns appreciable probability to responses that the teacher considers very unlikely, the log-ratio weight $w_\theta(y \mid x) = \log \pi_t(y \mid x) - \log \pi_\theta(y \mid x)$ can amplify OPD's gradient noise. Under specific conditions, OPD's gradient estimation variance can be proved to exceed that of sampled-response SFT by an arbitrarily large factor, see Appendix F.

These results are also supported by our experiments, see Appendix J for more details.

4 Variance Accumulation in Iterative Algorithms

Distillation algorithms have a fixed target distribution for the student to fit. Although we have showed in Theorem 2.1, the ideal single-step update of RL or MaxRL with binary reward will not affect the target distribution, this property may not hold in real applications. One important reason is that, there are non-negligible random noise in the one optimization step. This could be caused by the empirical estimator of the objective, the approximation of the optimizer and the numerical precision of the model parameter.

In this section we study *variance accumulation* in iterative training and its relation to the observed entropy dynamics, starting from the gradient noise in self-distillation.

4.1 Iterative Self-Distillation

We start from *iterative self-distillation*: each step samples responses from the current model and fits this empirical distribution. This is equivalent to an uncentered RL update with reward 1 for every response, and coincides with sampled-response SFT at the proximal limit each step. The distinction is that, in SFT at the proximal limit, the target distribution is fixed as $\pi_t = \pi_{\theta_0}$, whereas in iterative self-distillation the target distribution is π_θ which updates with the current policy.

By Lemma 3.1, the sampled gradient has zero mean but generally nonzero variance. Fix a prompt x, write $\pi_k = \pi_{\theta_k}(\cdot \mid x)$, and let $\mathcal{F}_k$ denote the training history before sampling at step k. Then we have the following result:

Lemma 4.1 (Martingale structure of self-distillation)**.** *For iterative self-distillation with step size η_k, the parameters form a martingale. The induced probabilities satisfy the same property to first order:*

$$\mathbb{E}[\theta_{k+1} \mid \mathcal{F}_k] = \theta_k, \qquad \mathbb{E}[\pi_{k+1}(y) \mid \mathcal{F}_k] = \pi_k(y) + O(\eta_k^2). \tag{10}$$

Proof: Appendix G.

theorem 4.1 (Conditional decrease of expected entropy)**.** *Let $h_x(\theta) = H(\pi_\theta(\cdot \mid x))$. If h_x is concave along every possible update segment from θ_k to θ_{k+1}, then*

$$\mathbb{E}[H(\pi_{k+1}) \mid \mathcal{F}_k] \leq H(\pi_k). \tag{11}$$

The inequality is strict under nonzero conditional update variance.

Proof: Appendix G.

Thus sampling noise can deterministicly reduce expected entropy even when the parameter update has zero mean, while iterative generation severely amplifies this effect. We should note that the iterative training is unavoidable in RL or self-distillation theoretically, since the distribution drift already occurs, the objective itself will require updating the actor's parameter or introducing importance sampling correction (Schulman et al., 2017b).

This expectational decrease of entropy should be understood in this way: We have no guarantee on the deterministic direction of entropy in each step. If one training step happens to generate some samples with low probability, self-distillation on these samples will actually increase the sample entropy. So this is not contradictive with previous researches that carefully analyzed the condition when a single learning step will increase or decrease the entropy (Cui et al., 2025; Ren & Sutherland, 2025; Jin et al., 2026; Wang et al., 2026). This result in fact takes a step further, and answers if we are doing the update multiple times, or in the long-term of training, how should the entropy change on average. The long-term expectation cancels the randomness in a single step update, and gives a deterministic decrement.

4.2 Reversed Iterative Self-Distillation

A common explanation for entropy collapse in large-model training is that high-probability samples are more likely to be sampled, and further increasing their probabilities leads to an additional reduction in entropy (Cui et al., 2025; Hao et al., 2026). This interpretation is consistent with our findings. In fact, the expected entropy decrease established by Theorem 11 further shows that, although low-probability samples may occasionally be sampled and increase the entropy during training, such increases are, in expectation, dominated by the overall decreasing trend.

To further illustrate this phenomenon, we additionally conduct a reversed iterative self-distillation experiment. In this setting, we assign a reward of -1 to all samples. As a result, high-probability samples are also more likely to be penalized, thereby tending to increase the entropy. Nevertheless, after reversing the sign of the objective, it still retains the properties of zero mean and nonzero variance, and the expected entropy decrease predicted by Theorem 11 continues to hold. This is a relatively counterintuitive result. It further predicts that the rare entropy-decreasing events under such training should be particularly pronounced, which is indeed what we observe empirically, as shown in Figure 1. More results can be found in Section J.3.

4.3 Variance Accumulation in RL

While the entropy collapse is widely observed in RL scenarios, previous experiments could provide certain evidence that the variance accumulation is also one of the main causes of this phenomena in RL. Another important side evidence of the variance impact in RL is the advantage estimation methods adopted in PPO (Schulman et al., 2017b), GRPO (Shao et al., 2024), or RLOO (Ahmadian et al., 2024). These methods can effectively lower down the variance of the estimator in RL algorithms. This notably increases the stability of RL training, while our deduction further indicates that they are can also prevent potential entropy collapse of the target distribution. It remains open if we could find better methods in controling the estimattion variance.

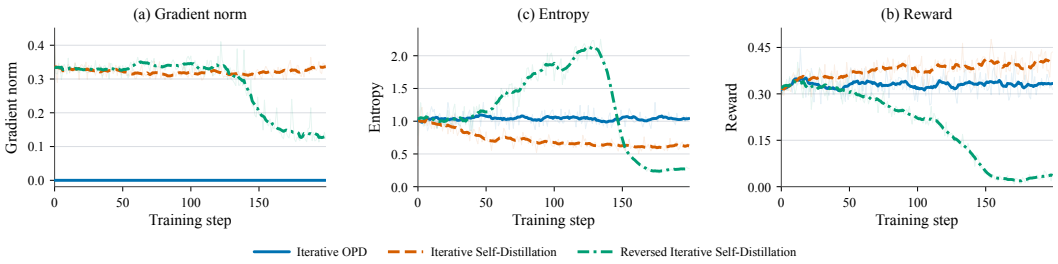

Figure 1: The comparision of the entropy dynamics of distillation algorithms. The nonzero variance and iteration are two necessasy conditions for variance accumulation to occur. RL is where both conditions are satisfied.

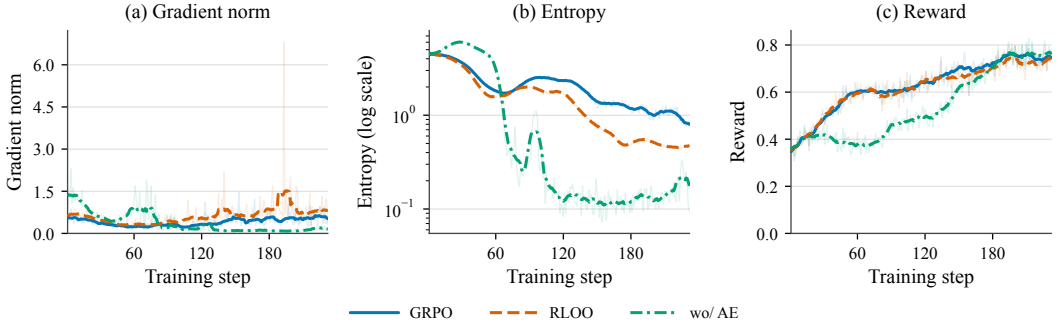

Figure 2: Ablation study on the impact of advantage estimation methods. We can see that under the same RL setting, advantage estimation methods constantly lowers the entropy deduction. Although they eventually achieved similar reward on the training set, to some extent, lower entropy still reflects reduced diversity and weaker generalization ability.

5 Conditional KL Regularization in RL

Insight. In iterative algorithms, the target distribution helps separate intended gains from unintended side effects. For example, RL aims to maximize expected reward. With binary rewards, conditioning on $r(x, y) = 1$ yields the training target π_θ^+, which remains unchanged under the ideal update in Theorem 2.1. Tracking how this target changes during training can therefore reveal side effects beyond the intended reward improvement.

5.1 Conditional Decomposition in RL

A sequence-level binary reward naturally decomposes the probability distribution induced by the original model into two components, with the true conditional distribution serving as the target distribution. The policy entropy also decomposes into the binary entropy of correctness and an accuracy-weighted average of the true and false conditional entropies.

Lemma 5.1 (Accuracy-weighted entropy decomposition)**.** *For a prompt x with deterministic binary rewards, write $\rho = \rho_\theta(x) \in (0, 1)$ for the accuracy and suppress conditioning on x. Let $H(\pi_\theta^+)$ and $H(\pi_\theta^-)$ denote the true conditional and false conditional entropies, respectively. Then*

$$H(\pi_\theta) = h_{\mathrm{b}}(\rho) + \rho H(\pi_\theta^+) + (1 - \rho) H(\pi_\theta^-), \tag{12}$$

where $h_{\mathrm{b}}(\rho) = -\rho \log \rho - (1 - \rho) \log(1 - \rho)$. If $H(\pi_\theta^+) < H(\pi_\theta^-)$, increasing accuracy while keeping both conditional distributions fixed will decrease their weighted average entropy.

Proof: Appendix H.

Typically in practice, both conditional entropies are much larger than the binary entropy $h_{\mathrm{b}}(\rho) \leq \log 2$, the total entropy is well approximated by their accuracy-weighted average. This conditional

is supported in our experiments, see Figure 3. Both conditional entropies are significantly larger than the binary entropy, while false conditional entropy is usually several times higher than true conditional entropy.

This property can, to some extent, account for the approximately linear relationship between entropy and accuracy observed in prior studies (Cui et al., 2025). However, a mere shift in probability density is insufficient to explain the reductions in diversity and pass@k performance reported in other studies (Yue et al., 2025; Tajwar et al., 2026). This accuracy-driven source of entropy reduction in RL need not reflect a loss of diversity among correct responses. The concern with diversity collapse is the degeneration of the target and conditional entropy.

This conditional decomposition induced by RL also offers a new perspective on regularization. To prevent model collapse and preserve generalization, a common practice is to introduce a KL-divergence constraint that keeps the trained model from deviating excessively from the original model (Schulman et al., 2017a;b; Ouyang et al., 2022). However, directly constraining the KL divergence can also limit improvements in reward. Under the conditional decomposition of RL, what we truly need to control is the degradation of the target distribution. This observation motivates a Conditional KL (CKL) regularization scheme that directly constrains changes in the target distribution.

This CKL regularization admits a simple policy-gradient estimator:

> **Insight.** The mean log probability-ratio between the current and reference models over positive samples captures the average probability gain on correct solutions. CKL then regularizes only the deviation of each positive sample from this average gain. This actually introduces a baseline into the KL penalty for positive samples, while imposing no constraint on negative samples.

theorem 5.1 (Centered sequence-k_1 estimator for CKL)**.** *Fix a prompt x and suppress its conditioning. Let π_θ be a differentiable positive policy and π_{ref} a fixed reference. Define $K(\theta) = D_{\mathrm{KL}}(\pi_\theta^+ \| \pi_{\mathrm{ref}}^+)$. Draw G independent responses $Y_i \sim \pi_{\theta_0}$, and let $I^+ = \{i : r(x, Y_i) = 1\}$ and $N = |I^+|$. The empirical signal is*

$$d_i = \log \frac{\pi_{\theta_0}(Y_i)}{\pi_{\mathrm{ref}}(Y_i)}, \qquad \widehat{c}_i = \begin{cases} d_i - \dfrac{1}{N}\sum_{j \in I^+} d_j, & i \in I^+,\ N \geq 2, \\ 0, & \textit{otherwise}. \end{cases}$$

Writing $s_i = \nabla_\theta \log \pi_\theta(Y_i)|_{\theta_0}$, the estimator satisfies, for every n with $\Pr(N = n) > 0$,

$$\mathbb{E}\left[\frac{1}{G} \sum_{i=1}^{G} \widehat{c}_i s_i \,\middle|\, N = n \right] = \frac{\max(n-1, 0)}{G}\, \nabla_\theta K(\theta)|_{\theta_0}\,. \tag{13}$$

Thus a detached reward $-\beta \widehat{c}_i$, with $\beta \geq 0$, gives an expected policy-gradient contribution proportional to the negative conditional KL gradient.

Proof: Appendix I.

As shown in Figure 3, we indeed observe that CKL effectively relaxes the constraint imposed by standard KL regularization on accuracy improvements while preserving the true conditional entropy. Although CKL cannot achieve completely lossless regularization due to cross-sequence interactions induced by model parameterization, it nevertheless demonstrates substantial practical value.

This advantage of CKL still holds for up to 20 training epochs, as shown in Figure 4. We then evaluated these models on different math datasets, where CKL regularization did exhibit higher performance and generalizability over KL of similar strength, See Figure 5. More related results could be found in Section K.

6 Conclusions

Distribution matching offers a new perspective on LLM training algorithms. The objective of a variety of LLM training algorithms can be unified as a divergence between the policy model and a target distribution. The target distribution itself also serves as an effective lens through which the properties of training algorithms can be analyzed.

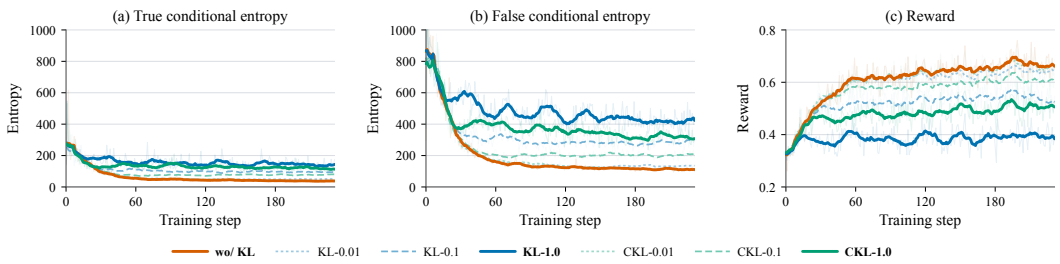

Figure 3: The comparison of KL and Conditional KL (CKL) regularization in RL. We trained Qwen3-1.7B-Base (Yang et al., 2025) on MATH (Hendrycks et al., 2021) dataset with GRPO (Shao et al., 2024) algorithm and different KL penalities. Two conditional entropies are measured on the sequence level. We can see that strong KL control strength severely limits the reward increment in RL, while CKL regularization allows the model to obtain higher reward while maintaining the diversity of true samples.

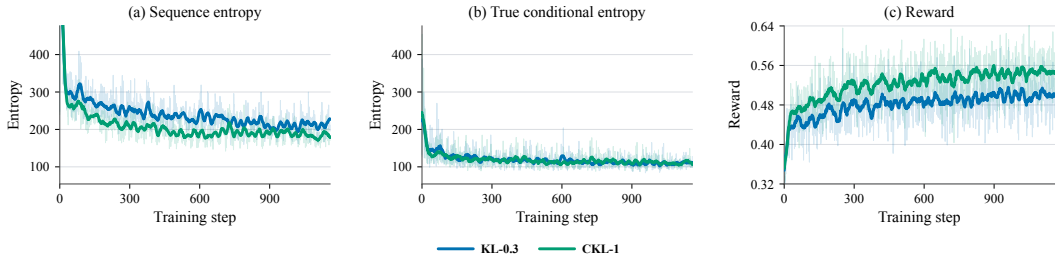

Figure 4: Extended experiments on KL and CKL regularization. These experiments follows the same setting as that in Figure 3. We can see that CKL-1 and KL-0.3 exhibits similar regularization ability on conditional distribution, while CKL-1 constantly obtains higher accuracy. Its entropy will thus be slightly lower according to the accuracy-weighted decomposition.

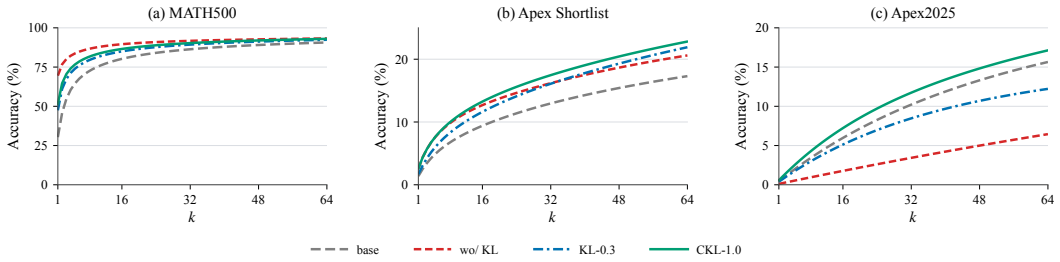

Figure 5: The comparison of pass@k performance on MATH500 (Lightman et al., 2023), Apex Shortlist (Balunović et al., 2025) and Apex2025 (Balunović et al., 2025). Models are trained for 20 epochs on MATH dataset. We can see that models trained without KL has higher accuracy on datasets similar to the training set, while severe collapse occurs on harder datasets like Apex2025. CKL constantly achieves higher generalizability over KL regularization across these experiments.

In this work, we show that forward KL distribution matching exhibits non-vanishing variance in the proximal limit, which provides a possible explanation for the advantage of OPD over SFT in proximal settings. Furthermore, for iterative algorithms such as RL, we find that the cumulative effect of this non-vanishing variance can lead to entropy collapse of the target distribution. This observation offers a potential explanation for the degradation of output diversity observed under prolonged RL training in related studies. Motivated by the role of the target distribution in RL, we further develop a Conditional KL regularization method and empirically demonstrate its advantages over classical KL regularization in language model. We hope that our findings can provide useful insights for future research and practical applications in this area.

AI USE STATEMENT

In this work, we used generative AI tools for figure illustration and polishment of the writing. We also used them to retrieve and summarize related works and modifying the training code. AI tools also helped us in early theoretical exploration. The ideation and evaluation were done by the authors. In paper writing we used AI tool to summarize and refine our drafts in theoretical exploration. We did not use generative AI tools for directly draft sections of the paper or generating synthetic datasets.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to Basics: Revisiting REINFORCE Style Optimization for Learning from Human Feedback in LLMs, February 2024. URL http://arxiv.org/abs/2402.14740. arXiv:2402.14740 [cs.LG].

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization, June 2019. URL http://arxiv.org/abs/1811.11214. arXiv:1811.11214 [cs.LG].

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM2: When Smol Goes Big – Data-Centric Training of a Small Language Model, February 2025. URL http://arxiv.org/abs/2502.02737. arXiv:2502.02737 [cs.CL].

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. MathArena: Evaluating LLMs on Uncontaminated Math Competitions, October 2025. URL http://arxiv.org/abs/2505.23281. arXiv:2505.23281 [cs].

Zhicheng Cai, Xinyuan Guo, Hanlin Wu, Mingxuan Wang, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Beyond Euclidean Clipping: Overcoming Exploration Collapse in LLM RL via Riemannian Isometric Policy Optimization, July 2026. URL http://arxiv.org/abs/2607.10169. arXiv:2607.10169 [cs.LG].

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems, November 2021. URL http://arxiv.org/abs/2110.14168. arXiv:2110.14168 [cs].

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models, May 2025. URL http://arxiv.org/abs/2505.22617. arXiv:2505.22617 [cs].

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng

Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *Nature*, 645(8081):633–638, September 2025. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://arxiv.org/abs/2501.12948. arXiv:2501.12948 [cs].

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J. Yang, J. Q. Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, M. S. Di, M. Y. Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, S. H. Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W. Zhang, W. L. Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, Y. C. Yan, Y. Q. Wang, Y. W. Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, Z. F. Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe

Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. DeepSeek-V4: Towards Highly Efficient Million-Token Context Intelligence, April 2026. URL http://arxiv.org/abs/2606.19348. arXiv:2606.19348 [cs.CL].

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. AReaL: A Large-Scale Asynchronous Reinforcement Learning System for Language Reasoning, November 2025. URL http://arxiv.org/abs/2505.24298. arXiv:2505.24298 [cs].

Chang Gao, Chujie Zheng, Xiong-Hui Chen, Kai Dang, Shixuan Liu, Bowen Yu, An Yang, Shuai Bai, Jingren Zhou, and Junyang Lin. Soft Adaptive Policy Optimization, December 2025. URL http://arxiv.org/abs/2511.20347. arXiv:2511.20347 [cs].

GLM-5-Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. GLM-5: from Vibe Coding to Agentic Engineering, February 2026. URL http://arxiv.org/abs/2602.15763. arXiv:2602.15763 [cs.LG].

Zhezheng Hao, Hong Wang, Haoyang Liu, Jian Luo, Jiarui Yu, Hande Dong, Qiang Lin, Can Wang, and Jiawei Chen. Rethinking Entropy Interventions in RLVR: An Entropy Change Perspective, April 2026. URL http://arxiv.org/abs/2510.10150. arXiv:2510.10150 [cs.LG].

Bingxiang He, Zekai Qu, Zeyuan Liu, Yinghao Chen, Yuxin Zuo, Cheng Qian, Kaiyan Zhang, Weize Chen, Chaojun Xiao, Ganqu Cui, Ning Ding, and Zhiyuan Liu. JustRL: Scaling a 1.5B LLM with a Simple RL Recipe, December 2025. URL http://arxiv.org/abs/2512.16649. arXiv:2512.16649 [cs].

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset, November 2021. URL http://arxiv.org/abs/2103.03874. arXiv:2103.03874 [cs].

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, March 2015. URL http://arxiv.org/abs/1503.02531. arXiv:1503.02531 [stat.ML].

Renren Jin, Pengzhi Gao, Yuqi Ren, Zhuowen Han, Tongxuan Zhang, Wuwei Huang, Wei Liu, Jian Luan, and Deyi Xiong. Revisiting Entropy in Reinforcement Learning for Large Reasoning Models, April 2026. URL http://arxiv.org/abs/2511.05993. arXiv:2511.05993 [cs].

Muhammad Khalifa, Hady Elsahar, and Marc Dymetman. A Distributional Approach to Controlled Text Generation, May 2021. URL http://arxiv.org/abs/2012.11635. arXiv:2012.11635 [cs.CL].

Vijay R. Konda and John N. Tsitsiklis. OnActor-Critic Algorithms. *SIAM Journal on Control and Optimization*, 42(4):1143–1166, January 2003. ISSN 0363-0129, 1095-7138. doi: 10.1137/S0363012901385691. URL https://epubs.siam.org/doi/10.1137/S0363012901385691.

Tomasz Korbak, Hady Elsahar, Germán Kruszewski, and Marc Dymetman. On Reinforcement Learning and Distribution Matching for Fine-Tuning Language Models with no Catastrophic Forgetting, November 2022a. URL http://arxiv.org/abs/2206.00761. arXiv:2206.00761 [cs.LG].

Tomasz Korbak, Ethan Perez, and Christopher L. Buckley. RL with KL penalties is better viewed as Bayesian inference, October 2022b. URL http://arxiv.org/abs/2205.11275. arXiv:2205.11275 [cs.LG].

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking On-Policy Distillation of Large Language Models: Phenomenology, Mechanism, and Recipe, April 2026. URL http://arxiv.org/abs/2604.13016. arXiv:2604.13016 [cs].

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step, May 2023. URL http://arxiv.org/abs/2305.20050. arXiv:2305.20050 [cs].

Anamika Lochab, Bolian Li, and Ruqi Zhang. Uniform-Correct Policy Optimization: Breaking RLVR's Indifference to Diversity, May 2026. URL http://arxiv.org/abs/2605.00365. arXiv:2605.00365 [cs.LG].

Minjae Oh, Sangjun Song, Gyubin Choi, Yunho Choi, and Yohan Jo. KL for a KL: On-Policy Distillation with Control Variate Baseline, May 2026. URL http://arxiv.org/abs/2605.07865. arXiv:2605.07865 [cs.LG].

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, March 2022. URL http://arxiv.org/abs/2203.02155. arXiv:2203.02155 [cs].

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 Technical Report, January 2025. URL http://arxiv.org/abs/2412.15115. arXiv:2412.15115 [cs].

Yi Ren and Danica J. Sutherland. Learning Dynamics of LLM Finetuning, June 2025. URL http://arxiv.org/abs/2407.10490. arXiv:2407.10490 [cs].

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust Region Policy Optimization, April 2017a. URL http://arxiv.org/abs/1502.05477. arXiv:1502.05477 [cs].

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, August 2017b. URL http://arxiv.org/abs/1707.06347. arXiv:1707.06347 [cs].

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models, April 2024. URL http://arxiv.org/abs/2402.03300. arXiv:2402.03300 [cs].

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum Likelihood Reinforcement Learning, February 2026. URL http://arxiv.org/abs/2602.02710. arXiv:2602.02710 [cs].

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, M. C, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y. Charles, H. S. Che, Guanduo Chen, Guangyu Chen, Guanzheng Chen, Huarong Chen, Jia Chen, Jianlong Chen, Jun Chen, Kexin Chen, Peng Chen, Ruijue Chen, Wentao Chen, Xin Chen, Yang Chen, Yanru Chen, Yifei Chen, Yingjiang Chen, Yuankun Chen, Yujie Chen, Yutian Chen, Zhirong Chen, Dazhi Cheng, Yean Cheng, Jialei Cui, Jingbing Cui, Anqi Dai, Jiaqi Deng, Hao Ding, Rui Ding, Shaofeng Ding, Mengfan Dong, Mengnan Dong, Yuhao Dong, Yuxin Dong, Angang Du, Chenzhuang Du, Dikang Du, Jusen Du, Yulun Du, Yu Fan, Jing Feng, Qiulin Feng, Yichen Feng, Kelin Fu, Qiang Fu, Fuxuan Gao, Hongcheng Gao, Jingyue Gao, Tong Gao, Weijia Gao, Shangyi Geng, Jie Gong, Linhu Gong, Shengao Gong, Xiaochen Gong, Qizheng Gu, Yicheng Gu, Shuhao Guan, Haiqing Guo, Shiqi Guo, Xiang Guo, Zhengyan Guo, Beixi Hao, Wenxin Hao, Xiaoru Hao, Dailan He, Haotian He, Lehan He, Qi He, Weiran He, Xinran He, Xinyi He, Yibo He, Yunjia He, Chao Hong, Tiange Hong, Hao Hu, Jiaxi Hu, Ruikun Hu, Weiming Hu, Yangyang Hu, Zhenxing Hu, Liang Hua, Jinbin Huang, Ke Huang, Ruiyuan Huang, Siying Huang, Weixiao Huang, Yan Huang, Zhengjie Huang, Zhiqi Huang, Yulong Hui, Chaobo Jia, Yutong Jiang, Zhejun Jiang, Zuoyou Jiang, Wenyi Jin, Xinyi Jin, Yu Jing, Huanjun Kong, Guokun Lai, Aidi Li, Cheng Li, Chengyuan Li, Cong Li, Fang Li, Guanyu Li, Haoyang Li, Jia Li, Junxiong Li, Lei Li, Letian Li, Lincan Li, Weihong Li, Wentao Li, Xintong Li, Yang Li, Yishen Li, Yiwei Li, Yuxiao Li, Zhaowei Li, Zhaoxi Li, Zheming Li, Zhengxiao Li, Zhiyuan Li, Jiawei Lin, Xiaohan Lin, Yibo Lin, Zichao Lin, Ziyan Lin, Bill Liu, Boxiao Liu, Chuan Liu, Liang Liu, Shaowei Liu, Shudong Liu, Shuran Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yanming Liu, Yibo Liu, Yipeng Liu, Zhengying Liu, Zhiheng Liu, Enzhe Lu, Haoyu Lu, Linqiang Lu, Tingzhan Lu, Zhiyuan Lu, Aotian Luo, G. Luo, Junyu Luo, Yifan Luo, B. Lyu, Wenzhou Lyu, Shaoguang Mao, Yuan Mei, Xin Men, Minqing Ni, Yixuan Niu, Siyuan Pan, Shujun Peng, Zhangyang Qi, Ruoyu Qin, ZeChao Qin, Zeyu Qin, Haiquan Qiu, Jianxin Qiu, Jiezhong Qiu, Bowen Qu, Yuhao Qu, Zeyu Shang, Youbo Shao, Han Shen, Jincheng Shi, Juanfeng Shi, Lidong Shi, Shengyuan Shi, Wingchun Siu, Pengwei Song, Xiaoxi Song, Jianlin Su, Yunfeng Su, Zhaochen Su, Lin Sui, Jingsong Sun, Junyao Sun, Shaoning Sun, Shuzhe Sun, Tongyu Sun, Yujun Sun, Yunpeng Tai, Chuning Tang, Heyi Tang, Sirui Tang, Zecheng Tang, Chaoran Tian, Rongpeng Tian, Yu Tian, Wei Tu, Chensi Wang, Chuang Wang, Chunjie Wang, Dinglu Wang, Feng Wang, Hailong Wang, Haiming Wang, Hao Wang, Hao Wang, Huaqing Wang, Hui Wang, Jiayi Wang, Jinglong Wang, Jinhong Wang, Jiuzheng Wang, Linian Wang, Shaobo Wang, Shenzhi Wang, Shuyi Wang, Si Wang, Siyuan Wang, Tianfu Wang, Wenjue Wang, Xingran Wang, Xinmei Wang, Xinyuan Wang, Xusheng Wang, Yalin Wang, Yangkun Wang, Yao Wang, Yaoyu Wang, Yejie Wang, Yiqin Wang, Yucheng Wang, Yuzhi Wang, Zhaoji Wang, Zhaowei Wang, Zhengtao Wang, Zhenhao Wang, Zhongsheng Wang, Zifan Wang, Chu Wei, Ming Wei, Shouxin Wei, Zichen Wen, Fan Wu, Haoning Wu, Rucong Wu, Wenhao Wu, Xiaoxue Wu, Yingcong Wu, Yongqi Wu, Yuxin Wu, Zijian Wu, Xinglang Xian, Chenxuan Xiang, Yuye Xiang, Bocheng Xiao, Chenjun Xiao, Xin Xiao, Jin Xie, Xiaotong Xie, Yifeng Xie, Zhe Xie, Bowei Xing, Yiming Xiong, Baosheng Xu, Boyu Xu, Jiale Xu, Jianfan Xu, Jing Xu, Jinjing Xu, L. H. Xu, Qingtao Xu, Shuyao Xu, Suting Xu, Tiantian Xu, Tianxiang Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ye Xu, Yueni Xu, Ziyao Xu, Haonan Xue, Junjie Yan, Yaoyao Yan, Fan Yang, Guangyao Yang, Hao Yang, Junwei Yang, Ruoyu Yang, Wenjie Yang, Xiaofei Yang, Xinyu Yang, Yi Yang, Yiling Yang, Ying Yang, Yuchen Yang, Zhen Yang, Zhilin Yang, Zian Yang, Zuhao Yang, Haotian Yao, Dan Ye, Haoran Ye, Wenjie Ye, Zhanbo Ye, Bohong Yin, Haoxiang Yin, Xietong Yin, Chengzhen Yu, Haozhen Yu, Longhui Yu, Shengnan Yu, Shuying Yu, Tianxiang Yu, Enming Yuan, Mengjie Yuan, Tongtian Yue, Wei Yue, Yang Yue, Dunyuan Zha, Haobing Zhan, B. H. Zhang, Dehao Zhang, Fei Zhang, Hao Zhang, Haoyuan Zhang, Huanyu Zhang, Jiapei Zhang, Jiaxuan Zhang, Jin Zhang, Kaiyi Zhang, Miaozhen Zhang, Puqi Zhang, Qinglei Zhang, Rong Zhang, Rui Zhang, Shaoshuai Zhang, Shiyi Zhang, Xiaobin Zhang, Xiaoyun Zhang, Y. Zhang, Yangkun Zhang, Ye Zhang, Yichi Zhang, Yikun Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang,

Yutao Zhang, Yutong Zhang, Zheng Zhang, Zijing Zhang, Bin Zhao, Chenguang Zhao, Feifan Zhao, Jinglun Zhao, Jinxiang Zhao, Shuai Zhao, Wenshuo Zhao, Xiangyu Zhao, Xuanle Zhao, Yikai Zhao, Zijia Zhao, Haozhi Zheng, Huabin Zheng, Ruihan Zheng, Shaojie Zheng, Tengyang Zheng, Haofeng Zhong, Lei Zhong, Longguang Zhong, M. Zhou, Qiankang Zhou, Runjie Zhou, Ruozhang Zhou, Xinyu Zhou, Yiqiao Zhou, Zaida Zhou, Jinguo Zhu, Liya Zhu, Xinhao Zhu, Yangjunfeng Zhu, Yuxuan Zhu, Zhen Zhu, Chen Zhuang, Weiyu Zhuang, and Xinxing Zu. Kimi K3: Open Frontier Intelligence, July 2026. URL http://arxiv.org/abs/2607.24653. arXiv:2607.24653 [cs.CL].

Thinking Machines Lab and Kevin Lu. On-Policy Distillation. Technical report, Thinking Machines Lab: Connectionism, October 2025. URL https://thinkingmachines.ai/blog/on-policy-distillation/.

Shumin Wang, Yuexiang Xie, Wenhao Zhang, Yuchang Sun, Yanxi Chen, Yaliang Li, and Yanyong Zhang. On the Entropy Dynamics in Reinforcement Fine-Tuning of Large Language Models, February 2026. URL http://arxiv.org/abs/2602.03392. arXiv:2602.03392 [cs].

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, May 1992. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00992696. URL http://link.springer.com/10.1007/BF00992696.

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An Open-Source LLM Reinforcement Learning System at Scale, March 2025. URL http://arxiv.org/abs/2503.14476. arXiv:2503.14476 [cs].

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model?, November 2025. URL http://arxiv.org/abs/2504.13837. arXiv:2504.13837 [cs.AI].

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group Sequence Policy Optimization, July 2025. URL http://arxiv.org/abs/2507.18071. arXiv:2507.18071 [cs].

Xiangxin Zhou, Zichen Liu, Haonan Wang, Chao Du, Min Lin, Chongxuan Li, Liang Wang, and Tianyu Pang. Variational Reasoning for Language Models, October 2025. URL http://arxiv.org/abs/2509.22637. arXiv:2509.22637 [cs].

A Related Works

Modern LLM training typically begins with pretraining on massive datasets, with the goal of modeling the natural language distribution and acquiring broad world knowledge. Mid-training generally follows the same underlying learning paradigm, while differences in training data and parameter configurations can lead to substantially different outcomes. In the post-training stage, a common practice is to train specialized expert models for tasks in different domains, typically using SFT for cold start followed by task-specific RL algorithms. These expert models can then be distilled into a unified model through methods such as OPD, resulting in a final general-purpose model (DeepSeek-AI et al., 2026; Team et al., 2026; GLM-5-Team et al., 2026).

A.1 Distribution Matching Perspective

Among the algorithms used across different stages of LLM training, most admit a relatively clear interpretation from the perspective of distribution matching, with RL being a notable exception. Despite its distinctive optimization formulation, RL has also been widely recognized as closely related to distribution matching (Korbak et al., 2022a). Prior to our work, a number of studies investigated the connection between RL and distribution-matching objectives and derived locally optimal solutions with an exponential-tilting form similar to that in our Theorem 2.1 (Korbak et al., 2022b;a). Reward-induced conditional distributions have likewise been studied in prior work and have motivated several algorithmic improvements (Tajwar et al., 2026; Zhou et al., 2025).

In most previous studies, however, such distributions primarily appeared as auxiliary results in the derivation or analysis of RL objectives. Subtle discrepancies remain between RL optimization and direct distribution matching, preventing the two perspectives from being fully unified (Korbak et al., 2022a). Recent studies on MaxRL provide an important step toward bridging this gap, revealing a first-order approximation between RL and distribution matching. More importantly, our work differs in that we treat the target distribution induced by distribution matching as a central object of study, rather than merely an intermediate analytical construct. In particular, we study both its role in estimating the training objective and its stochastic evolution throughout training, which together constitute the main results of this work. We expect that this perspective may provide a useful lens for further understanding and improving related training algorithms.

A.2 Entropy Collapse in RL

Diversity collapse and entropy decrement in reinforcement learning have long been recognized in the classical RL literature. RL optimization can drive a model toward a suboptimal solution with reduced behavioral diversity, thereby limiting the model's long-term training potential. Classical approaches to mitigating this issue include entropy regularization (Ahmed et al., 2019) and constraints based on KL divergence (Schulman et al., 2017a). These methods, however, introduce their own trade-offs, including optimization instability and restrictions on reward improvement.

As RL has become increasingly prevalent in LLM post-training, a growing body of work has revisited entropy collapse in this setting. Existing studies have investigated its causes and potential remedies from a variety of perspectives, including biases introduced by PPO clipping (Yu et al., 2025; Cai et al., 2026; Jin et al., 2026), geometric biases induced by model architecture (Ren & Sutherland, 2025), and the underlying training dynamics (Wang et al., 2026). In contrast, our analysis identifies a long-horizon degradation of the target distribution caused by variance accumulation. This phenomenon arises at a more fundamental level and does not depend on a particular RL algorithm or implementation trick. Based on this analysis, we further derive a effective regularization strategy. Entropy collapse may represent one of the central challenges in scaling post-training, and our results provide an additional perspective on its underlying mechanism.

A.3 Variance Control and Training Stability

The stability of RL algorithms has been studied extensively, with the high variance of training-objective estimators being one of the primary challenges. A substantial fraction of algorithmic developments in RL can be understood as attempts to address this issue (Konda & Tsitsiklis, 2003; Schulman et al., 2017a;b). Advantage estimation is among the most widely adopted approaches: it can

be implemented using a separate critic model (Konda & Tsitsiklis, 2003; Schulman et al., 2017a;b), or approximated through techniques such as group-based sampling and normalization (Shao et al., 2024; Ahmadian et al., 2024). Beyond advantage estimation, the treatment of importance weights and the design of the underlying training infrastructure can also have a substantial impact on optimization stability (Zheng et al., 2025; Gao et al., 2025).

Recent empirical evidence suggests that, in the LLM setting, the strong prior induced by extensive pretraining can enable stable RL training over a meaningful range even in the absence of aggressive variance-reduction techniques (Ahmadian et al., 2024). Nevertheless, controlling training variance can still yield measurable performance gains. Our analysis further provides a possible explanation for these gains: reducing training variance not only stabilizes individual optimization steps, but can also mitigate the degradation of the training target caused by variance accumulation over time. As a result, the model is more likely to preserve meaningful exploration and reach better regions of the parameter space.

There are also many other works studying the stability issue and improvements of OPD (Li et al., 2026; Oh et al., 2026), while a most simple yet effective method could be a two-stage distillation, which uses SFT as coldstart, and use OPD to further improve their performance (Yang et al., 2025). Our analysis could provide more theoretical insights on the distillation algorithms as well.

B Limitation

In the previous sections we proposed a framework that unifies different training algorithms via distribution matching objective. For RL algorithms we only analyzed the simplest 0-1 binary reward setting. For more generalized multi-value reward RL we can also extend the distribution matching by introducing a randomized acceptance event, see Appendix D. However, a detailed investigation of its properties is left for future work.

A more important limitation of this work is that, we did not consider generalization ability of these methods. The main theoretical results derived in this paper apply directly to the behavior on the training set, while a more complete understanding of the corresponding behavior on the test set requires further theoretical analysis. Nevertheless, many of our experiments are also conducted on the test set, where we empirically observe behavior similar to that on the training set.

C Proof of Reward Reweighting Under a Forward-KL Restriction

> **Insight.** The forward-KL penalty makes the optimum interior and unique, so a single Lagrange multiplier determines how each response probability is reweighted. Responses with the same reward receive the same multiplier, which cancels when conditioning on a reward level set.

Fix a prompt x and a finite response space. Write $p_y = \pi_\theta(y \mid x) > 0$, $r_y = r(x, y) \in \mathbb{R}$, and $\beta > 0$.

Proof of Theorem 2.1. On the probability simplex,

$$F(q) = \sum_y q_y r_y - \beta D_{\mathrm{KL}}(p \| q) \tag{C.1}$$

has Hessian $-\beta \operatorname{diag}(p_y/q_y^2) \prec 0$ and tends to $-\infty$ at the boundary. Its unique maximizer is interior, with Lagrange conditions

$$r_y + \frac{\beta p_y}{q_y} - \lambda_x = 0 \quad \Longrightarrow \quad q_y^* = \frac{\beta p_y}{\lambda_x - r_y}, \qquad \sum_y \frac{\beta p_y}{\lambda_x - r_y} = 1. \tag{C.2}$$

The normalization sum decreases continuously from $+\infty$ to 0 on $(\max_y r_y, \infty)$, determining a unique λ_x. For any nonempty $\mathcal{S}_a = \{y : r_y = a\}$,

$$q^*(y \mid \mathcal{S}_a) = \frac{p_y \beta/(\lambda_x - a)}{\sum_{z \in \mathcal{S}_a} p_z \beta/(\lambda_x - a)} = p(y \mid \mathcal{S}_a), \qquad y \in \mathcal{S}_a. \tag{C.3}$$

□

Binary rewards. For $r_y \in \{0, 1\}$ and $0 < \rho_\theta(x) < 1$, the same solution is

$$q_y^* = \frac{p_y e^{\alpha_x r_y}}{1 - \rho_\theta(x) + \rho_\theta(x) e^{\alpha_x}}, \qquad \alpha_x = \log \frac{\lambda_x}{\lambda_x - 1} > 0. \tag{C.4}$$

Thus $q^*(\mathcal{S}_1) > \rho_\theta(x)$ and $(q^*)^+ = \pi_\theta^+$. For constant rewards, $q^* = p$.

KL constraint. Set $\delta_\beta = D_{\mathrm{KL}}(p\|q^*)$. For any q with $D_{\mathrm{KL}}(p\|q) \le \delta_\beta$, optimality gives

$$\mathbb{E}_q[r] \le \mathbb{E}_{q^*}[r] + \beta\big(D_{\mathrm{KL}}(p\|q) - \delta_\beta\big) \le \mathbb{E}_{q^*}[r], \tag{C.5}$$

with equality only at $q = q^*$.

D Proof of the True Conditional KL Identity and Its MaxRL Connection

> **Insight.** Conditioning on success rescales every successful response by the same factor, so the KL reduces to the negative logarithm of the success probability. Differentiating this probability and using the score identity then gives the success-weighted MaxRL gradient.

Let $\mathcal{D}$ be a fixed empirical prompt distribution. For each x, assume a finite response space, a differentiable positive policy π_θ, and a fixed binary reward. Write $\mathcal{S}_x^+ = \{y : r(x, y) = 1\}$ and assume $\rho_\theta(x) > 0$.

Proof of Theorem 2.2. Since $\pi_\theta^+(y \mid x) = \pi_\theta(y \mid x)/\rho_\theta(x)$ on $\mathcal{S}_x^+$ and vanishes elsewhere,

$$D_{\mathrm{KL}}(\pi_\theta^+ \| \pi_\theta) = \sum_{y \in \mathcal{S}_x^+} \pi_\theta^+(y \mid x) \log \frac{1}{\rho_\theta(x)} = -\log \rho_\theta(x). \tag{D.1}$$

Taking the negative expectation over x proves Eq. equation 8. Moreover,

$$\nabla_\theta \log \rho_\theta(x) = \frac{\sum_y r(x, y) \nabla_\theta \pi_\theta(y \mid x)}{\rho_\theta(x)} = \mathbb{E}_{y \sim \pi_\theta^+(\cdot|x)}[\nabla_\theta \log \pi_\theta(y \mid x)], \tag{D.2}$$

and hence

$$\nabla_\theta \mathcal{J}_{\mathrm{MaxRL}}^{(\infty)} = \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(\cdot|x)} \left[\frac{r(x, y)}{\rho_\theta(x)} \nabla_\theta \log \pi_\theta(y \mid x) \right]. \tag{D.3}$$

□

Frozen target. For $q_x = \pi_{\theta_0}^+(\cdot \mid x)$,

$$D_{\mathrm{KL}}(q_x \| \pi_\theta) = D_{\mathrm{KL}}(q_x \| \pi_\theta^+) - \log \rho_\theta(x). \tag{D.4}$$

The first term has zero gradient at θ_0, so

$$-\nabla_\theta \mathbb{E}_{x \sim \mathcal{D}} D_{\mathrm{KL}}(q_x \| \pi_\theta)|_{\theta_0} = \nabla_\theta \mathcal{J}_{\mathrm{MaxRL}}^{(\infty)}\Big|_{\theta_0}. \tag{D.5}$$

Finite-order MaxRL. Using $\log \rho = -\sum_{k \ge 1} (1 - \rho)^k / k$ for $0 < \rho \le 1$, the order-K objective of Tajwar et al. (2026) satisfies

$$\mathcal{J}_{\mathrm{MaxRL}}^{(K)} = -\mathbb{E}_{x \sim \mathcal{D}} \left[\sum_{k=1}^{K} \frac{(1 - \rho_\theta(x))^k}{k} \right], \tag{D.6}$$

$$\nabla_\theta \mathcal{J}_{\mathrm{MaxRL}}^{(K)} = \mathbb{E}_{x \sim \mathcal{D}} \left[\frac{1 - (1 - \rho_\theta(x))^K}{\rho_\theta(x)} \nabla_\theta \rho_\theta(x) \right]. \tag{D.7}$$

The geometric sum gives the gradient, and $1/k \le 1/(K + 1)$ for $k > K$ gives

$$0 \le \mathcal{J}_{\mathrm{MaxRL}}^{(K)} - \mathcal{J}_{\mathrm{MaxRL}}^{(\infty)} \le \mathbb{E}_{x \sim \mathcal{D}} \left[\frac{(1 - \rho_\theta(x))^{K+1}}{(K + 1)\rho_\theta(x)} \right]. \tag{D.8}$$

Thus $\mathcal{J}_{\mathrm{MaxRL}}^{(1)} = \mathcal{J}_{\mathrm{RL}} - 1$. Finite prompt support and $\rho_\theta(x) > 0$ imply convergence of the objective and gradient to Eqs. equation 8 and equation D.3 as $K \to \infty$.

D.1 Extension to Nonbinary Rewards via Randomized Acceptance

Insight. Rescaling a bounded reward to $[0, 1]$ allows it to serve as an acceptance probability. Conditioning on acceptance extends the MaxRL identity to the joint response–acceptance distribution; standard RL is recovered at first order.

Under the same policy assumptions, let the fixed reward satisfy $a \le r(x, y) \le b$, with constants $a < b$, and set $s(x, y) = (r(x, y) - a)/(b - a)$. Introduce $A \mid x, Y \sim \text{Bernoulli}(s(x, Y))$, with kernel $K_s(1 \mid x, y) = s(x, y)$, $K_s(0 \mid x, y) = 1 - s(x, y)$. Define

$$P_\theta(y, A \mid x) = \pi_\theta(y \mid x) K_s(A \mid x, y), \qquad \rho^s_\theta(x) = P_\theta(A = 1 \mid x) = \mathbb{E}_{\pi_\theta(\cdot|x)}[s(x, Y)]. \tag{D.9}$$

Assume $\rho^s_\theta(x) > 0$ and condition on $A = 1$:

$$Q_\theta(y, A \mid x) = P_\theta(y, A \mid x, A = 1) = \frac{\pi_\theta(y \mid x) s(x, y)}{\rho^s_\theta(x)} \mathbf{1}\{A = 1\}. \tag{D.10}$$

Since $Q_\theta / P_\theta = 1/\rho^s_\theta(x)$ on the support of Q_θ,

$$-\mathbb{E}_{x\sim\mathcal{D}} D_{\mathrm{KL}}\big(Q_\theta(\cdot, \cdot \mid x) \| P_\theta(\cdot, \cdot \mid x)\big) = \mathbb{E}_{x\sim\mathcal{D}}\left[\log \mathbb{E}_{\pi_\theta(\cdot|x)}[s(x, Y)]\right]. \tag{D.11}$$

For the response marginal $q^s_\theta(y \mid x) = \pi_\theta(y \mid x) s(x, y)/\rho^s_\theta(x)$, the KL chain rule gives

$$D_{\mathrm{KL}}(Q_\theta \| P_\theta) = D_{\mathrm{KL}}(q^s_\theta \| \pi_\theta) + \mathbb{E}_{q^s_\theta}[-\log s(x, Y)]. \tag{D.12}$$

The expectation is over $s(x, y) > 0$. For binary r with $s = r$, the extra term vanishes, recovering Eq. equation 8. Finally, $\log \rho = (\rho - 1) + O((1 - \rho)^2)$ near $\rho = 1$ gives standard RL on s at first order, and $\mathbb{E}[s] = (\mathbb{E}[r] - a)/(b - a)$.

Table 2: Continuation of Table 1: bounded nonbinary rewards. Objectives are averaged over prompts $x \sim \mathcal{D}$. The model distribution is P_θ; the target Q_θ conditions it on acceptance and has $Q_\theta(y, 0 \mid x) = 0$. † RL is its first-order approximation.

Algorithm	Objective function	Target distribution	Divergence
Iterative algorithms			
RL	$\mathbb{E}_{\pi_\theta}[s(x, Y)] = \rho^s_\theta(x)$	$Q_\theta(y, 1 \mid x) = \dfrac{s(x, y)\pi_\theta(y \mid x)}{\rho^s_\theta(x)}$	Forward KL†
MaxRL (extended)	$\log \mathbb{E}_{\pi_\theta}[s(x, Y)] = -D_{\mathrm{KL}}(Q_\theta \| P_\theta)$	$Q_\theta(y, 1 \mid x) = \dfrac{s(x, y)\pi_\theta(y \mid x)}{\rho^s_\theta(x)}$	Forward KL

E Proof of Gradient Noise at the Proximal Limit

Insight. When student and teacher coincide, every OPD log-ratio weight is zero, whereas the sampled KD scores cancel only in expectation.

Fix a prompt x, a finite response space, and a fixed positive teacher π_t. Let $\pi_\theta > 0$ be continuously differentiable near θ_0, with $\pi_{\theta_0} = \pi_t$. Write $s_\theta(y \mid x) = \nabla_\theta \log \pi_\theta(y \mid x)$ and $w_\theta(y \mid x) = \log[\pi_t(y \mid x)/\pi_\theta(y \mid x)]$. For $m \ge 1$, define the sampled-response estimators

$$\widehat{g}_{\mathrm{KD}}(\theta) = \frac{1}{m}\sum_{i=1}^{m} s_\theta(Y_i \mid x), \qquad Y_i \overset{\text{iid}}{\sim} \pi_t(\cdot \mid x), \tag{E.1}$$

$$\widehat{g}_{\mathrm{OPD}}(\theta) = \frac{1}{m}\sum_{i=1}^{m} w_\theta(\widetilde{Y}_i \mid x) s_\theta(\widetilde{Y}_i \mid x), \qquad \widetilde{Y}_i \overset{\text{iid}}{\sim} \pi_\theta(\cdot \mid x). \tag{E.2}$$

Proof of Lemma 3.1. Since $\mathbb{E}_{\pi_\theta}[s_\theta] = \nabla_\theta \sum_y \pi_\theta(y \mid x) = 0$,

$$\nabla_\theta[-D_{\mathrm{KL}}(\pi_t \| \pi_\theta)] = \mathbb{E}_{\pi_t}[s_\theta], \tag{E.3}$$

$$\nabla_\theta[-D_{\mathrm{KL}}(\pi_\theta \| \pi_t)] = \mathbb{E}_{\pi_\theta}[(w_\theta - 1)s_\theta] = \mathbb{E}_{\pi_\theta}[w_\theta s_\theta], \tag{E.4}$$

so both estimators are unbiased. At θ_0, $w_{\theta_0} = 0$ and $\mathbb{E}_{\pi_t}[s_{\theta_0}] = 0$. Therefore $\widehat{g}_{\mathrm{OPD}}(\theta_0) = 0$ almost surely, while independence gives

$$\mathrm{Cov}(\widehat{g}_{\mathrm{KD}}(\theta_0)) = \frac{1}{m}\mathbb{E}_{\pi_t}[s_{\theta_0} s_{\theta_0}^\top] = \frac{1}{m} F_x(\theta_0). \tag{E.5}$$

For $A \in \{\mathrm{KD}, \mathrm{OPD}\}$, set $V_A(\theta) = \mathbb{E}\|\widehat{g}_A(\theta) - \mathbb{E}\widehat{g}_A(\theta)\|_2^2$. Then

$$\begin{aligned} mV_{\mathrm{KD}}(\theta) &= \mathbb{E}_{\pi_t}\|s_\theta\|_2^2 - \|\mathbb{E}_{\pi_t} s_\theta\|_2^2, \\ mV_{\mathrm{OPD}}(\theta) &= \mathbb{E}_{\pi_\theta}[w_\theta^2 \|s_\theta\|_2^2] - \|\mathbb{E}_{\pi_\theta}[w_\theta s_\theta]\|_2^2. \end{aligned} \tag{E.6}$$

These finite sums are continuous in θ. If $\mathrm{Tr}\, F_x(\theta_0) > 0$, then $V_{\mathrm{OPD}}(\theta_0) = 0 < V_{\mathrm{KD}}(\theta_0)$, so the strict inequality persists in a neighborhood of θ_0. □

F A Conditional Variance Comparison in the Distal Regime

Insight. Suppressing teacher probability on a set still sampled by the student introduces a growing $\log(1/\epsilon)$ term in the OPD weights. Separating this term from a bounded remainder reveals quadratic growth of OPD variance under the stated nonzero-score condition, while KD variance remains bounded by the fixed student scores.

Fix x, a finite response space $\mathcal{Y}_x$, and a positive policy π_θ continuously differentiable near $\bar{\theta}$. Write $p_y = \pi_{\bar{\theta}}(y \mid x)$ and $s_y = \nabla_\theta \log \pi_\theta(y \mid x)|_{\bar{\theta}}$. For a nonempty proper subset $\mathcal{A}$ of $\mathcal{Y}_x$, set $a = \sum_{y \in \mathcal{A}} p_y \in (0, 1)$. Given fixed positive distributions u on $\mathcal{A}$ and v on its complement, define

$$\pi_{t,\epsilon}(y \mid x) = \begin{cases} \epsilon u_y, & y \in \mathcal{A}, \\ (1-\epsilon) v_y, & y \notin \mathcal{A}, \end{cases} \qquad 0 < \epsilon < 1. \tag{F.1}$$

With m independent responses and teacher $\pi_{t,\epsilon}$, let $V_A(\epsilon) = \mathrm{Tr}\,\mathrm{Cov}(\widehat{g}_A(\bar{\theta}))$ for the estimators in Eqs. equation E.1–equation E.2.

Lemma F.1 (Variance amplification under teacher-probability suppression)**.** *If $\sum_{y \in \mathcal{A}} p_y \|s_y\|_2^2 > 0$, then, as $\epsilon \to 0$,*

$$V_{\mathrm{OPD}}(\epsilon) = \frac{C_{\mathcal{A}}}{m} \log^2 \frac{1}{\epsilon} + O\left(\frac{\log(1/\epsilon)}{m}\right), \qquad V_{\mathrm{KD}}(\epsilon) \le \frac{G^2}{m}, \tag{F.2}$$

where $G = \max_y \|s_y\|_2$ and

$$C_{\mathcal{A}} = \sum_{y \in \mathcal{A}} p_y \|s_y\|_2^2 - \left\| \sum_{y \in \mathcal{A}} p_y s_y \right\|_2^2 > 0. \tag{F.3}$$

Consequently, for every fixed $R > 0$, sufficiently small ϵ gives $V_{\mathrm{OPD}}(\epsilon) > R V_{\mathrm{KD}}(\epsilon)$.

Proof. Set $L = \log(1/\epsilon)$ and $w_\epsilon(y) = \log[\pi_{t,\epsilon}(y \mid x)/p_y]$. Then

$$w_\epsilon(y) = -L\mathbf{1}_{\mathcal{A}}(y) + b_\epsilon(y), \qquad b_\epsilon(y) = \begin{cases} \log(u_y/p_y), & y \in \mathcal{A}, \\ \log(v_y/p_y) + \log(1-\epsilon), & y \notin \mathcal{A}. \end{cases} \tag{F.4}$$

For $0 < \epsilon \le 1/2$, b_ϵ and s are uniformly bounded. Expanding the variance under $Y \sim p$ gives

$$mV_{\mathrm{OPD}}(\epsilon) = L^2 \,\mathrm{Tr}\,\mathrm{Cov}_p(\mathbf{1}_{\mathcal{A}}(Y) s_Y) + O(L) = L^2 C_{\mathcal{A}} + O(L). \tag{F.5}$$

Cauchy–Schwarz yields $C_{\mathcal{A}} \ge (1-a)\sum_{y \in \mathcal{A}} p_y \|s_y\|_2^2 > 0$. For KD, $mV_{\mathrm{KD}}(\epsilon) \le \mathbb{E}_{\pi_{t,\epsilon}}\|s_Y\|_2^2 \le G^2$, which proves Eq. equation F.2 and the comparison for any fixed $R > 0$. Also, $D_{\mathrm{KL}}(p\|\pi_{t,\epsilon}) = aL + O(1) \to \infty$. □

G Martingales and Entropy in Self-Distillation

Insight. Sampling from the current policy makes the score update have conditional mean zero, so the parameters form a martingale despite nonzero sampling noise.

For Lemmas 4.1–4.1, fix a prompt x and a finite response space $\mathcal{Y}_x$. Assume a positive, twice continuously differentiable policy, integrable iterates and increments, and $\mathcal{F}_k$-measurable step sizes η_k and batch sizes $m_k \geq 1$. With the teacher frozen within each step, plain SGD gives

$$\theta_{k+1} = \theta_k + \eta_k \widehat{g}_k, \qquad \widehat{g}_k = \frac{1}{m_k} \sum_{i=1}^{m_k} s_{\theta_k}(Y_{k,i} \mid x), \qquad Y_{k,i} \mid \mathcal{F}_k \overset{\text{iid}}{\sim} \pi_k, \tag{G.1}$$

where $\pi_k(y) = \pi_{\theta_k}(y \mid x)$ and $s_\theta = \nabla_\theta \log \pi_\theta$.

Proof of Lemma 4.1. The score identity and conditional independence yield

$$\mathbb{E}[\widehat{g}_k \mid \mathcal{F}_k] = 0, \qquad \operatorname{Cov}(\theta_{k+1} - \theta_k \mid \mathcal{F}_k) = \frac{\eta_k^2}{m_k} F_x(\theta_k), \tag{G.2}$$

so $\mathbb{E}[\theta_{k+1} \mid \mathcal{F}_k] = \theta_k$. If $M_{k,y}$ bounds the operator norm of the probability Hessian along the update segments, Taylor expansion gives

$$|\mathbb{E}[\pi_{k+1}(y) \mid \mathcal{F}_k] - \pi_k(y)| \leq \frac{M_{k,y}\eta_k^2}{2m_k} \operatorname{Tr} F_x(\theta_k) = O(\eta_k^2), \tag{G.3}$$

locally as $\eta_k \to 0$ for fixed θ_k. □

For the entropy result, let $h_x(\theta) = -\sum_y \pi_\theta(y \mid x) \log \pi_\theta(y \mid x)$ and $d_k = \theta_{k+1} - \theta_k$. Assume that, for every possible update d_k and $t \in [0, 1]$,

$$d_k^\top \nabla_\theta^2 h_x(\theta_k + t d_k) d_k \leq -\mu_k \|d_k\|_2^2, \qquad \mu_k \geq 0, \tag{G.4}$$

where μ_k is $\mathcal{F}_k$-measurable. The case $\mu_k = 0$ is concavity along the update segments; $\mu_k > 0$ gives strict decrease whenever the update variance is nonzero.

Insight. Mean-zero parameter noise can lower expected entropy when entropy is concave along the possible update segments. In the Taylor expansion, the linear term vanishes in conditional expectation, leaving a nonpositive curvature term whose magnitude is bounded using the Fisher covariance.

Proof of Theorem 4.1. Taylor's formula with equation G.4, $\mathbb{E}[d_k \mid \mathcal{F}_k] = 0$, and equation G.2 yields

$$\mathbb{E}[H(\pi_{k+1}) \mid \mathcal{F}_k] \leq H(\pi_k) - \frac{\mu_k \eta_k^2}{2m_k} \operatorname{Tr} F_x(\theta_k) \leq H(\pi_k). \tag{G.5}$$

Taking expectations proves the claim, with strict decrease if the subtracted term is positive with positive probability. □

H Proof of the Accuracy-Weighted Entropy Decomposition

Fix a prompt x, a finite response space, and a deterministic binary reward. Write $\rho = \rho_\theta(x) \in (0, 1)$, $p = \pi_\theta(\cdot \mid x)$, and $p^\pm = \pi_\theta^\pm(\cdot \mid x)$. Then

$$p^+(y) = \frac{r(x, y)p(y)}{\rho}, \qquad p^-(y) = \frac{(1 - r(x, y))p(y)}{1 - \rho}. \tag{H.1}$$

Proof of Lemma 5.1. Since p^+ and p^- have disjoint supports and $p = \rho p^+ + (1 - \rho)p^-$, using $0 \log 0 = 0$ gives

$$\begin{aligned} H(p) &= -\rho \sum_y p^+(y) \log\big(\rho p^+(y)\big) - (1 - \rho) \sum_y p^-(y) \log\big((1 - \rho)p^-(y)\big) \\ &= h_{\mathrm{b}}(\rho) + \rho H(p^+) + (1 - \rho)H(p^-), \end{aligned}$$

which proves Eq. equation 12. For fixed p^+ and p^- with $H(p^+) < H(p^-)$,

$$\frac{\mathrm{d}}{\mathrm{d}\rho}\big[\rho H(p^+) + (1-\rho)H(p^-)\big] = H(p^+) - H(p^-) < 0.$$

□

I Proof of the Conditional KL Estimator

Insight. On successful responses, the sequence log-ratio is the sum of a success-probability log-ratio and a conditional-distribution log-ratio. Centering within successes removes the former exactly. Multiplying the remaining signal by the policy score gives the conditional-KL gradient. Using the same samples to estimate the mean introduces a finite-group scaling factor, with zero signal for groups containing fewer than two successes.

Fix a prompt x and a finite response space with a deterministic binary reward and a nonempty success set $\mathcal{S}^+$. Write $p_\theta = \pi_\theta(\cdot \mid x)$, $p = p_{\theta_0}$, $q = \pi_{\mathrm{ref}}(\cdot \mid x)$, $\rho_\theta = p_\theta(\mathcal{S}^+)$, $\rho = \rho_{\theta_0}$, and $\rho_q = q(\mathcal{S}^+)$. Assume p_θ is differentiable and both p_θ and q are positive. Superscript $+$ denotes conditioning on $\mathcal{S}^+$; all gradients below are evaluated at θ_0.

Proof of Theorem 5.1. Let $d(y) = \log(p(y)/q(y))$, $\mu = \mathbb{E}_{p^+}[d(Y)]$, and $K(\theta) = D_{\mathrm{KL}}(p_\theta^+ \| q^+)$. For $y \in \mathcal{S}^+$,

$$d(y) = \log\frac{\rho}{\rho_q} + \log\frac{p^+(y)}{q^+(y)}, \qquad d(y) - \mu = \log\frac{p^+(y)}{q^+(y)} - K(\theta_0). \tag{I.1}$$

The scores $s = \nabla \log p_\theta$ and $s^+ = \nabla \log p_\theta^+$ satisfy $s = s^+ + \nabla \log \rho_\theta$ on $\mathcal{S}^+$ and $\mathbb{E}_{p^+}[s^+] = 0$. Hence

$$\begin{aligned}\nabla K &= \mathbb{E}_{p^+}\left[s^+(Y)\left(\log\frac{p^+(Y)}{q^+(Y)} + 1\right)\right] \\ &= \mathbb{E}_{p^+}[s^+(Y)(d(Y)-\mu)] = \mathbb{E}_{p^+}[s(Y)(d(Y)-\mu)].\end{aligned} \tag{I.2}$$

Conditional on $N = n \geq 2$, relabel the successful responses as $Y_1, \ldots, Y_n \overset{\text{iid}}{\sim} p^+$. Set $\bar{d} = n^{-1}\sum_{j=1}^n d(Y_j)$, $a = \mathbb{E}_{p^+}[d(Y)s(Y)]$, and $b = \mathbb{E}_{p^+}[s(Y)]$. Independence gives

$$\begin{aligned}\mathbb{E}[\bar{d}\, s(Y_i) \mid N = n] &= \frac{a}{n} + \frac{n-1}{n}\mu b, \\ \mathbb{E}[(d(Y_i) - \bar{d})s(Y_i) \mid N = n] &= \frac{n-1}{n}(a - \mu b) = \frac{n-1}{n}\nabla K.\end{aligned} \tag{I.3}$$

Summing over the n successes and dividing by G proves Eq. equation 13; for $n \leq 1$, both sides are zero. Multiplication by $-\beta$ gives the stated expected gradient contribution. Finally, $(d_i + a_0) - (\bar{d} + a_0) = d_i - \bar{d}$ proves invariance under a common shift of the successful log-ratios. □

J Supplementary Experiments on Proximal and Distal Distillation

We report supplementary results for distillation from identical, nearby, and more distant teachers.

J.1 Experimental settings

The proximal-limit experiment initializes both student and frozen teacher from Qwen3-1.7B-Base (Yang et al., 2025). Nearby teachers are obtained by training the same model with GRPO for 50, 100, or 200 steps. The distal experiments use a JustRL teacher with R1-Distill-1.5B (DeepSeek-AI et al., 2025) and Qwen2.5-Math-1.5B (Qwen et al., 2025) students; both use the same cached teacher responses for SFT. Training uses slime with Megatron and SGLang at revision `a067ce6f`, with:

- seed: 1;
- Qwen3 experiments: DAPO17k (Yu et al., 2025), maximum response length 2,048;
- JustRL experiments: JustRL training data (He et al., 2025), maximum response length 8,192;
- learning rate: 10^{-5}.

Distillation from nearby teachers. With each of the three GRPO-trained teachers, OPD achieves a higher mean evaluation score than SFT (Table 3), consistent with its proximal advantage in these single-seed experiments.

Distillation with a larger initial performance gap. With the JustRL teacher, both methods bring R1-Distill-1.5B close to teacher performance, with SFT slightly ahead. For Qwen2.5-Math-1.5B, OPD rollout correctness declines and remains low (Figure 6), while SFT improves the mean evaluation score from 15.27 to 47.46 (Table 3).

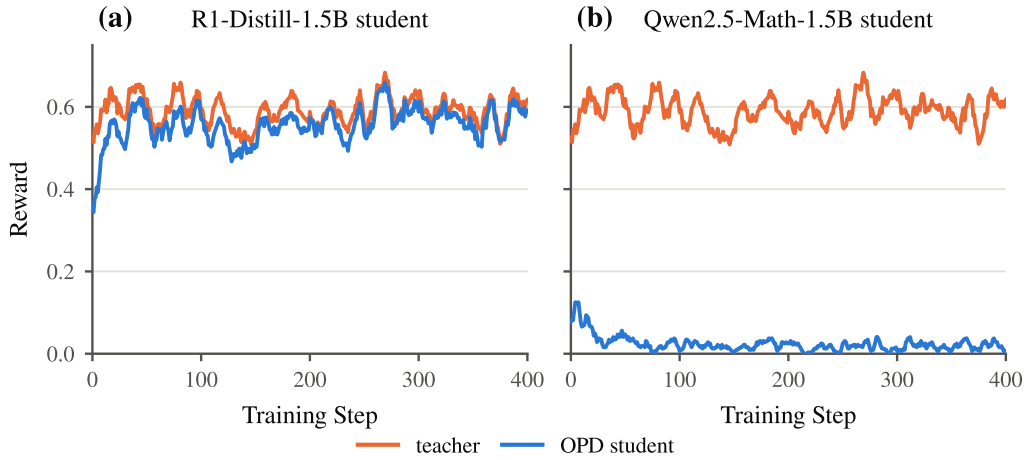

Figure 6: Reward curve of distal OPD with the JustRL teacher and two initial students. OPD rollout correctness increases for R1-Distill-1.5B but decreases and remains low for Qwen2.5-Math-1.5B. Note that since we do not have student's on policy samples in SFT, we can not draw a similar reward curve for SFT. Their evaluation results are reported in Table 3.

J.2 EVALUATION RESULTS

Table 3: Evaluation results of OPD and SFT with 16 sampled responses per problem. We can see that OPD typically outperforms SFT in proximal distillation, while it may experience severe instability issue and performance degradation in distal distillation. Proximal teachers are obtained by training the base model with GRPO for 50, 100, or 200 steps.

Student	Teacher	Model	AIME-24	AMC23	MATH500	Mean
Proximal Distillation						
Qwen3-1.7B-Base	GRPO-50	Student	0.62	7.50	16.19	8.10
		Teacher	1.04	12.19	25.62	12.95
		SFT	**1.88**	15.78	32.69	16.78
		OPD	1.25	**16.25**	**35.80**	**17.77**
Qwen3-1.7B-Base	GRPO-100	Student	0.62	7.50	16.19	8.10
		Teacher	**1.67**	18.28	33.50	17.82
		SFT	1.04	**19.06**	39.32	19.81
		OPD	0.62	**19.06**	**42.58**	**20.75**
Qwen3-1.7B-Base	GRPO-200	Student	0.62	7.50	16.19	8.10
		Teacher	2.50	22.50	46.27	23.76
		SFT	**3.33**	25.16	49.73	26.07
		OPD	2.92	**25.62**	**51.36**	**26.63**
Distal Distillation						
R1-Distilled-Qwen-1.5B	JustRL-DS	Student	21.04	63.75	75.90	53.56
		Teacher	**49.38**	83.28	84.16	72.27
		SFT	47.92	**85.16**	**85.74**	**72.94**
		OPD	47.29	83.44	85.39	72.04
Qwen2.5-Math-1.5B	JustRL-DS	Student	4.38	18.44	23.00	15.27
		Teacher	**49.38**	**83.28**	**84.16**	**72.27**
		SFT	11.88	56.09	74.41	47.46
		OPD	0.62	3.75	5.17	3.18

J.3 SUPPLIMENTARY EXPERIMENTS ON STANDARD AND REVERSED ITERATIVE SELF-DISTILLATION

Here are some supplimentary experiments on two iterative self-distillation methods mentioned in the main body, See Figure 7.

In a single training step, model entropy may randomly increase or decrease, but over the long run it exhibits a deterministic downward trend in expectation. These entropy curves satisfies the derivation in the main body very well. In fact the decreasing expected entropy does not guarantee the probability of decrement and increment. For example in reversed iterative self-distillation, since the punishment on samples with a high probability will raise the entropy, the entropy policy model has a high probability to be increased. However, the negative expectation indicates that once the decrement happens, the magnitude of the decrease will be substantial. This was also observed in our experiments.

These experiments share these experimental settings:

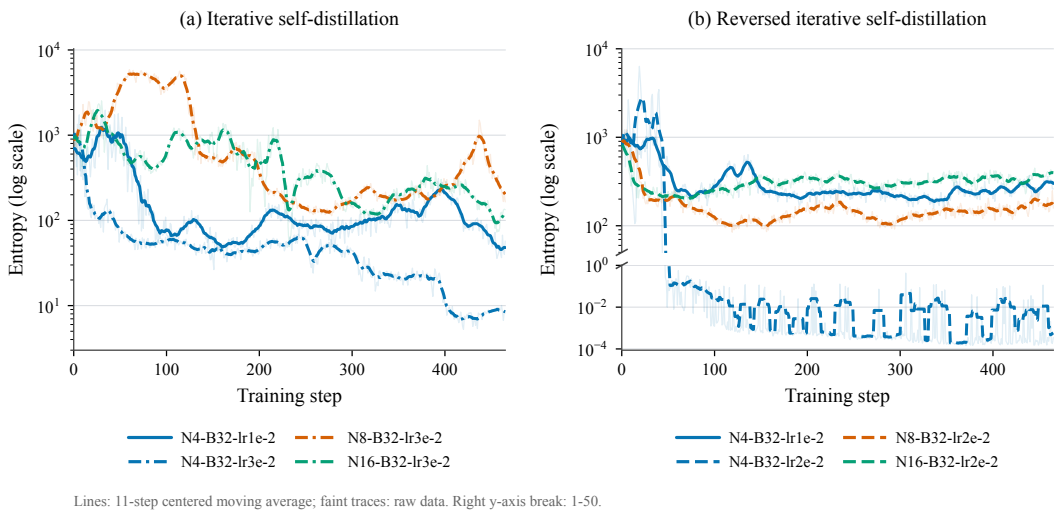

Figure 7: Additional experiments on entropy dynamics of iterative self-distillation and reversed iterative self-distillation. In the legend "N" denotes the number of samples per request, "B" denotes batchsize and "lr" denotes the learning rate of SGD optimizer.

- seed: 1
- total_train_epochs: 2
- actor.path: Qwen/Qwen3-1.7B-Base
- dataset: openai/gsm8k
- actor.dtype: bfloat16
- actor.kl_ctl: 0
- actor.optimizer.type: sgd
- actor.optimizer.momentum: 0

Other hyperparameters are kept the same as the default value of AReaL, which can be found in our codebase.

K Supplimentary Experiments on Reinforcement Learning

K.1 Additional Ablation Study on Advantage Estimation

Here are some other supplimentary experiments on RL absent in the main body. They provide more evidence on the theoretical results obtained by our analysis and beyond.

In Figure 8 we also examined the variance effect on smaller models like SmolLM2-360M-Instruct. We used similar training settings like that in Tajwar et al. (2026). We can see that advantage estimation methods effectively prevents entropy collapse and stabilizes the training process. These experiments share the following hyperparameter settings in AReaL (Fu et al., 2025), unmentioned hyperparameters are kept the same as the default setting:

- seed: 1
- batchsize: 128
- n_samples: 4
- total_train_steps: 1200
- actor.optimizer.lr: 1e-5
- actor.reward_scaling: 2.0
- actor.reward_bias: -0.5

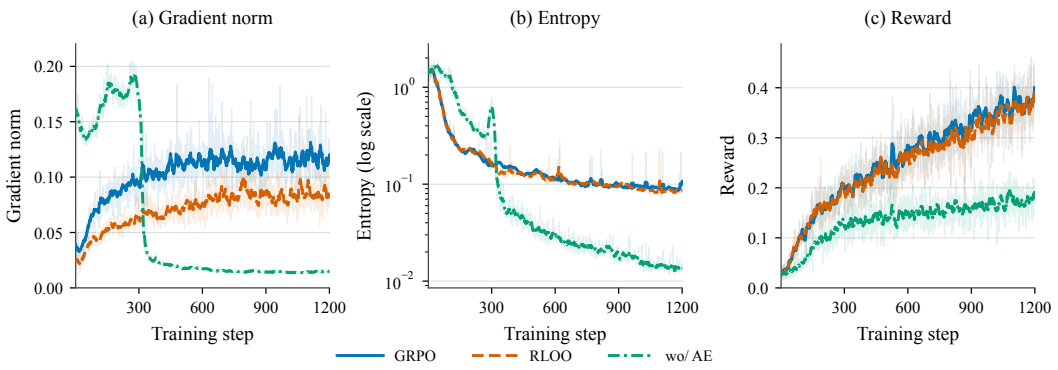

Figure 8: The ablation study on the advantage estimation on SmolLM2-360M-Instruct (Allal et al., 2025) and GSM8K (Cobbe et al., 2021) dataset. with batchsize 128, nsamples 4 and AdamW optimizer with learning rate 1e-5.

K.2 Additional Results on Conditional KL Restriction

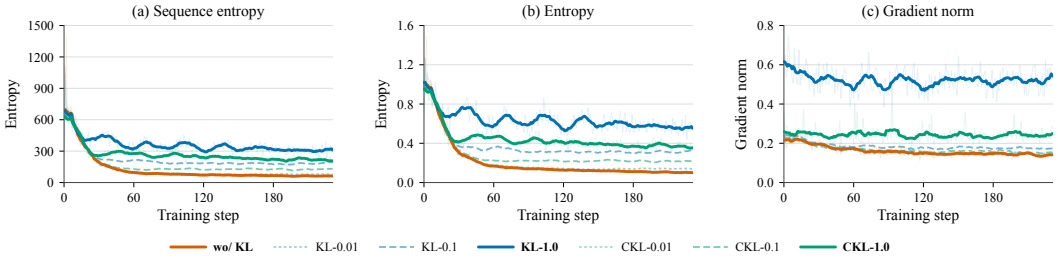

Figure 9: Additional metrics on the comparison of KL and CKL restrictions.

The results in Figure 9 come from the same set of experiments as that in Figure 3. We can see that the sequence-level entropy showed the same decreasing trend as that of token-level entropy. Recall the entropy decomposition in Lemma 5.1, the entropy of a model in RL is only comparable if they have similar accuracy. Otherwise the entropy value may not directly reflect the collapse in diversity. The gradient norm behavior of these experiments also provides some explanation on the advantage of CKL restriction. With high restriction strength, the gradient of CKL is much more modest while it still preserves the diversity very well.

These experiments adopts these hyperparameter settings, while unmentioned ones use the default value in AReaL (Fu et al., 2025):

- seed: 1
- batchsize: 128
- n_samples: 4
- total_train_epochs: 4
- actor.optimizer.lr: 1e-6
- actor.reward_scaling: 2.0
- actor.reward_bias: -0.5
- actor.kl_estimator: “k1”

L Further Discussions

Our analysis of variance accumulation concerns entropy evolution during prolonged language-model training. More broadly, it raises the possibility that stochastic fluctuations can induce a deterministic

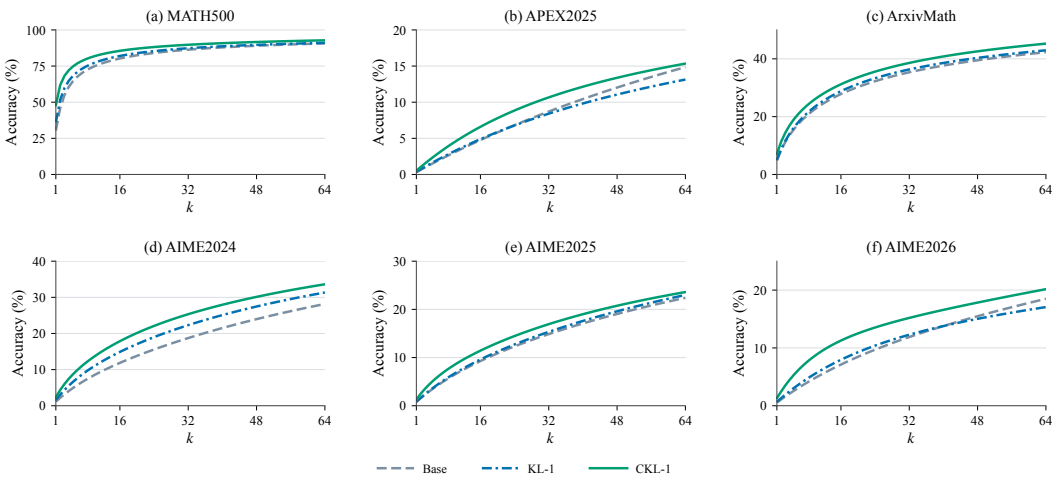

Figure 10: Pass@k performance of models trained for 4 epochs with KL or CKL regularization on MATH500 (Lightman et al., 2023), Apex2025 (Balunović et al., 2025), arxivmath (Balunović et al., 2025), and AIME (Balunović et al., 2025) datasets. CKL regularization also preserves more learning potential in the sense of pass@k performance compared with KL regularization with the same strength. The evaluation settings is the same as that used in AReaL training pipeline.

drift toward lower diversity in other iterative systems, including biological evolution and technological development. For example, even when two technological solutions are equally effective, random fluctuations in their early exploration and adoption could become self-reinforcing, leaving one insufficiently explored and producing a less diverse outcome. Such a mechanism could contribute to path dependence without requiring an intrinsic difference in the quality of the alternatives.

These connections remain speculative. Establishing whether variance accumulation operates in such systems, and distinguishing its effects from other mechanisms, would require domain-specific model and empirical evidence that are difficult to obtain and beyond the scope of this work. We mention these possibilities only to suggest broader directions for investigating the phenomenon.