TOWARDS UNDERSTANDING ON-POLICY DISTILLATION THROUGH THE LENS OF TEST-TIME SCALING

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) has emerged as a promising post-training technique for enhancing LLM reasoning. Under the reverse KL objective, the idealized optimum of OPD aligns the student distribution with that of the teacher. When the teacher consistently outperforms the student, this naturally suggests that OPD should yield broad improvements over the pre-OPD student. However, do such improvements extend across the entire range of test-time sampling budgets? In this work, we revisit this expectation through the lens of test-time scaling by varying the sampling budget K and evaluating performance with pass@K. Across multiple settings, we observe two distinct patterns: OPD can improve pass@K at both small and large sampling budgets, but it can also improve small-budget performance while reducing large-budget pass@K. We show one condition that guarantees such a reversal and an idealized reverse KL counterexample where it occurs even when the teacher has higher accuracy on every problem. To choose between two candidate teachers at a target sampling budget, we propose the *Teacher Advantage Score at K* (TAS@K), which can be computed before OPD training to predict which teacher will lead to a larger improvement in pass@K. Across three domains and thirteen benchmarks, the ordering predicted by TAS@K agrees with the observed pass@K improvements of the resulting OPD models in 83.6% of experiments, providing a useful signal for teacher selection at the target pass@K.

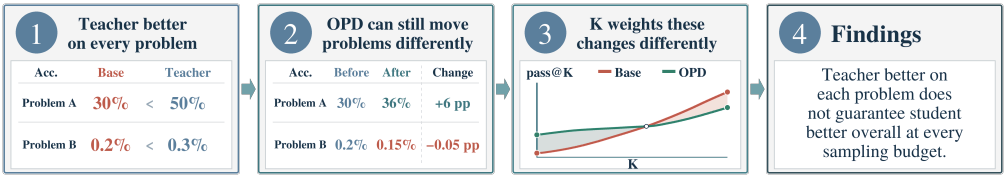


Figure 1: **Overview of the pass@K reversal in OPD.** Even when the teacher is better on every problem, OPD updates may improve the accuracy of some problems while slightly hurting others. Such uneven changes can increase pass@K at small K while decreasing it as K grows.

1 INTRODUCTION

On-policy distillation (OPD) has emerged as a promising approach for improving the reasoning performance of large language models (LLMs) (Agarwal et al., 2024; Song & Zheng, 2026), with its practical effectiveness demonstrated by recent industry-scale efforts, including DeepSeek-V4 (DeepSeek-AI et al., 2026), Qwen3 (Yang et al., 2025), and Nemotron-Cascade 2 (Yang et al., 2026d). In OPD, the student generates on-policy rollouts, while a stronger teacher provides token-level guidance through a reverse KL, encouraging the student distribution to move closer to the teacher (Li et al., 2026). Current evaluations primarily focus on the performance gains of OPD under relatively small test-time sampling budgets (e.g., pass@1) (Li et al., 2026; Jin et al., 2026; Yang et al., 2026b). This raises a natural question: as a distillation paradigm that explicitly aligns the student with a stronger teacher, does OPD yield consistent gains as the test-time sampling budget scales from small to large K?

Across multiple configurations in Table 1, we observe two qualitatively different patterns. As shown in Table 2, in some cases, OPD improves pass@K across the full range of sampling budgets. In others, it improves pass@K at small K while degrading performance at large K relative to the pre-OPD

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.

Table 1: **OPD settings.** Student-teacher pairs, benchmarks, and maximum sampling budgets.

Domain	Student	Teacher	Max K	Benchmarks
Math	Qwen3-1.7B-Base (Yang et al., 2025)	Qwen3-4B-Base-GRPO (Li et al., 2026)	1,024	AMC23 (math-ai, 2025),
	DS-Distill-Qwen-1.5B (Guo et al., 2025)	Skywork-OR1-Math-7B (He et al., 2025b)	1,024	AIME24 (HuggingFaceH4, 2025),
	DS-Distill-Qwen-1.5B (Guo et al., 2025)	JustRL-DS-1.5B (He et al., 2025a)	1,024	AIME25 (MathArena, 2025),
	Qwen3-1.7B (Yang et al., 2025)	Qwen3-4B-RL-Math (Yang et al., 2026b)	1,024	AIME26 (MathArena, 2026)
Code	Qwen3-1.7B (Yang et al., 2025)	Qwen3-8B (Yang et al., 2025)	512	HumanEval+, MBPP+ (Liu et al., 2023),
	Qwen3-1.7B (Yang et al., 2025)	Qwen3-4B-RL-Code (Yang et al., 2026b)	128	LiveCodeBench (Jain et al., 2025)
Fact			1,024	HotpotQA (Yang et al., 2018),
	Qwen3-1.7B (Yang et al., 2025)	Qwen3-8B-RL-RAR (Chen et al., 2025)	1,024	TriviaQA (Joshi et al., 2017),
	Qwen3-1.7B (Yang et al., 2025)	SESA-8B-search (Fu et al., 2026b)	1,024	PopQA (Mallen et al., 2023),
			1,024	2WikiMultiHopQA (Ho et al., 2020)

student. We further observe this latter pattern across multiple OPD variants, including EOPD (Jin et al., 2026), ExOPD (Yang et al., 2026b), and Direct-OPD (Feng et al., 2026), suggesting that the phenomenon is not specific to a particular OPD formulation. We give a sufficient condition under which pass@1 rises while pass@K falls at a larger budget. Our theoretical analysis in Section 3.3 shows that this can happen even when the teacher has higher accuracy on every problem.

Motivated by these observations, we further ask whether one can predict, before full OPD training, which of two teachers will produce the student with higher pass@K at a target sampling budget. To this end, we propose the *Teacher Advantage Score at K* (TAS@K), which compares the relative likelihood preference of the two candidate teachers for correct over incorrect responses generated by the pre-OPD base model, and weights each problem for the target budget. The sign of TAS@K provides a useful signal for selecting the teacher expected to yield the larger gain in target pass@K after OPD training. Among all 177 comparisons, TAS@K selects the teacher whose OPD-trained model achieves higher observed pass@K in 148 cases, achieving an agreement rate of 83.6%.

Our contributions are summarized as follows:

- We systematically study OPD through the lens of test-time scaling, showing that its gains can exhibit qualitatively different patterns across sampling budgets (Section 4).
- We theoretically provide a sufficient condition for a budget-dependent pass@K reversal, showing it can occur even when the teacher has higher accuracy on every problem (Section 3).
- We propose TAS@K, a teacher-selection score that can be computed before OPD training to predict which candidate teacher will yield a larger pass@K improvement after OPD (Section 5).

2 Preliminaries

2.1 On-Policy Distillation

Let π_θ and π_T denote the student and teacher models, respectively. Given a problem x, the student samples a reasoning trajectory y with length L. At the t-th decoding step, the student reaches the prefix state $(x, y_{<t})$, where the teacher provides a dense target distribution over the next token. The standard OPD objective minimizes the reverse KL divergence between the student and teacher distributions over these student-visited states (Li et al., 2026; Gu et al., 2024):

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)}\left[\sum_{t=1}^{L} D_{\mathrm{KL}}(\pi_\theta(\cdot \mid x, y_{<t}) \,\|\, \pi_T(\cdot \mid x, y_{<t}))\right]. \tag{1}$$

Here, $D_{\mathrm{KL}}(\cdot\|\cdot)$ denotes the KL divergence over the next-token vocabulary; the student distribution is its first argument and the teacher distribution is the second, yielding the reverse KL direction. In practice, OPD can use full-vocabulary KL, top-k token KL, or sampled-token estimators. Our settings include both top-k and sampled-token implementations. The defining distinction from off-policy distillation is the source of training trajectories: OPD samples them from the student policy being updated, so teacher feedback is based on the student's current behavior (Agarwal et al., 2024).

2.2 Sampling-Budget Evaluation

Test-time scaling allocates additional computation by sampling multiple responses for the same problem (Brown et al., 2024). Given a model π_θ and K independently sampled responses $\{y_i\}_{i=1}^K$ for a problem x, pass@K measures whether at least one response is correct (Chen et al., 2021):

$$\text{pass@}K = \mathbb{E}_{x\sim\mathcal{D},\,\{y_i\}_{i=1}^K\sim\pi_\theta(\cdot|x)}\left[\mathbb{I}(\exists i \in \{1,\ldots,K\},\ \text{s.t. } y_i \text{ is correct})\right]. \tag{2}$$

Pass@K reflects the model's ability to discover at least one successful reasoning trajectory with increased sampling opportunities. Small K measures how readily a model produces a correct response under a limited budget; large K indicates the capability coverage when more responses are sampled. More details of finite-sample estimator used in our experiments are provided in Appendix B.

3 Theoretical Analysis of Pass@K

To understand how OPD gains can persist or reverse as the sampling budget grows, we first examine how problem-level accuracy changes contribute to pass@K. We then ask whether a teacher that is more accurate on every problem prevents the student's accuracy from decreasing on any problem during OPD training. Let $p_\theta(x)$ denote the probability of a correct response to the problem x under the model π_θ, which we call its problem-level accuracy. For K independent responses sampled from the same model, we define the problem-level and dataset-level pass@K values as follows:

$$\text{pass@}K(x;\theta) = 1-(1-p_\theta(x))^K, \qquad \text{pass@}K(\theta) = \mathbb{E}_x[\text{pass@}K(x;\theta)].$$

3.1 Problem-Level Accuracy Changes Across Sampling Budgets

Let θ_0 and θ_T denote the student parameters before and after OPD. For a problem x, write $\Delta p(x) = p_{\theta_T}(x) - p_{\theta_0}(x)$ and $\Delta\text{pass@}K(x) = \text{pass@}K(x;\theta_T) - \text{pass@}K(x;\theta_0)$.

Lemma 1 (Problem-Level Pass@K Preserves the Direction of Accuracy Changes)**.** For every finite positive integer K, $\Delta\text{pass@}K(x)$ and $\Delta p(x)$ have the same sign. For fixed K, as $\Delta p(x) \to 0$,

$$\Delta\text{pass@}K(x) = K(1-p_{\theta_0}(x))^{K-1}\Delta p(x) + O(\Delta p(x)^2). \tag{3}$$

The direction of the change is exact for arbitrary finite changes in p, whereas Eq. (3) only characterizes its local magnitude. In this local approximation, $K(1-p_{\theta_0}(x))^{K-1}$ serves as a budget-dependent sensitivity factor that scales each problem's accuracy change. Thus, the same problem cannot improve at small K and degrade at large K, while for fixed $K > 1$, changes on lower-accuracy problems have a larger effect on pass@K. The exact finite-change identity and proof are provided in Appendix A.1.

3.2 Dataset-Level Gains and Losses

Since the problem-level accuracy changes $\Delta p(x)$ are fixed across sampling budgets, while the factor $K(1-p_{\theta_0}(x))^{K-1}$ that scales their contributions varies with K, the aggregate effect can change across budgets. At the dataset level, $\Delta\text{pass@}K = \mathbb{E}_x[\Delta\text{pass@}K(x)]$. Therefore, changes on different problems can be reweighted as K varies, allowing the overall gain to reverse sign. The following proposition gives a sufficient condition for such a reversal.

Proposition 1 (A Sufficient Condition for Higher Pass@1 but Lower Pass@K)**.** Let $G = \mathbb{E}_x[\max\{\Delta p(x), 0\}]$ and $L = \mathbb{E}_x[\max\{-\Delta p(x), 0\}] > 0$. Suppose every improved problem has initial accuracy at least p_+, and every degraded problem has initial accuracy at most p_-, where $0 < p_- < p_+ < 1$. For any fixed $K > 1$, the following inequality is sufficient for a reversal:

$$1 < \frac{G}{L} < \left(\frac{1-p_-}{1-p_+}\right)^{K-1} \quad\Longrightarrow\quad \Delta\text{pass@}1 > 0, \quad \Delta\text{pass@}K < 0. \tag{4}$$

Here, p_+ is a lower bound on the initial accuracy of improved problems, while p_- is an upper bound for degraded problems. At $K = 1$, $G > L$ raises pass@1; at the target K, the losses outweigh the gains after budget-dependent weighting. This condition is sufficient, not necessary. The proof, intermediate bound, and a numerical example are provided in Appendix A.1.

Table 2: **Pass@1 and full-budget pass@K (%) for the primary settings.** Pass@1 improves in every setting, while the full-budget pass@K change splits between gains and losses.

Domain	Student	Teacher	Pass@1 (Base → OPD)	Δ (pp)	Pass@K (Base → OPD)	Δ (pp)
Math	Qwen3-1.7B-Base	Qwen3-4B-Base-GRPO	10.6 → 18.4	+7.8	73.3 → 65.4	−7.9
	DS-Distill-Qwen-1.5B	Skywork-OR1-Math-7B	36.5 → 41.7	+5.1	87.5 → 83.5	−4.0
	DS-Distill-Qwen-1.5B	JustRL-DS-1.5B	36.5 → 51.8	+15.3	87.5 → 85.0	−2.5
	Qwen3-1.7B	Qwen3-4B-RL-Math	19.7 → 47.3	+27.6	79.2 → 83.5	+4.3
Code	Qwen3-1.7B	Qwen3-8B	36.6 → 38.8	+2.3	52.3 → 59.5	+7.2
	Qwen3-1.7B	Qwen3-4B-RL-Code	36.6 → 42.7	+6.1	52.3 → 70.1	+17.8
Fact	Qwen3-1.7B	Qwen3-8B-RL-RAR	9.5 → 10.5	+1.0	29.1 → 38.0	+8.9
	Qwen3-1.7B	SESA-8B-search	9.5 → 10.7	+1.2	29.1 → 37.6	+8.5

3.3 A Stronger Teacher Does Not Guarantee Problem-Level Improvement

We model small-step training by a differentiable path $\{\theta_\tau\}_{\tau\in[0,T]}$. For a student-visited prefix $s = (x', y_{<t}) \sim \rho_{\theta_\tau}$, consider full-vocabulary reverse KL as an idealized OPD update. Under a continuous-time approximation of the expected update, with sampled prefixes held fixed when computing gradients, let $b_x(\tau) = \frac{d}{d\tau} p_{\theta_\tau}(x)$ denote the rate at which accuracy changes on x:

$$b_x(\tau) = \mathbb{E}_{\substack{s\sim\rho_{\theta_\tau} \\ a\sim\pi_{\theta_\tau}(\cdot|s)}} \left[\log \frac{\pi_T(a \mid s)}{\pi_{\theta_\tau}(a \mid s)} \left\langle \nabla_\theta p_{\theta_\tau}(x), \nabla_\theta \log \pi_{\theta_\tau}(a \mid s) \right\rangle \right]. \tag{5}$$

Teacher preferences guide token updates at student-visited prefixes, but those updates can raise or lower problem-level accuracy. Update conventions and derivations are given in Appendices A.2-A.3.

Proposition 2 (A Stronger Teacher Does Not Prevent a Local Budget Reversal)**.** For any sampling budget J, let $\Delta_h \text{pass@}J = \text{pass@}J(\theta_h) - \text{pass@}J(\theta_0)$. For any fixed $K > 1$, suppose the initial OPD-induced accuracy rates on a finite empirical dataset satisfy

$$\mathbb{E}_x[b_x(0)] > 0, \qquad \mathbb{E}_x\left[K(1 - p_{\theta_0}(x))^{K-1} b_x(0)\right] < 0. \tag{6}$$

Then every sufficiently small $h > 0$ yields $\Delta_h \text{pass@}1 > 0$ and $\Delta_h \text{pass@}K < 0$. At $K = 1024$, these inequalities are realizable under full-vocabulary reverse KL distillation even when the teacher is more accurate on every problem; problem-level teacher and student accuracies alone therefore cannot tell whether this OPD update raises or lowers accuracy on a given problem.

Appendix A.4 gives the two-problem construction, and Appendix A.1 gives the finite-path identity. Exactly matching a teacher that is more accurate on every evaluation problem would improve all budgets, but does not imply monotonic improvement during finite training. Thus, small-budget gains can coexist with worse large-K performance; this is a possible outcome, not an inevitable limitation.

4 Empirical Results under Test-Time Scaling

4.1 Experimental Settings

OPD settings. Table 1 summarizes a series of OPD training settings and benchmarks used in this work, spanning three domains, twelve benchmarks, and eight student-teacher configurations. The Math experiments use DAPO-Math-17K (Yu et al., 2025), the Code experiments use CodeR1-12K (ganler, 2025), and the Fact experiments use training data from NQ (Kwiatkowski et al., 2019) and HotpotQA (Yang et al., 2018). More training details are provided in Appendix C.

Evaluation. For Math, we evaluate on AMC2023 (math-ai, 2025), AIME2024 (HuggingFaceH4, 2025), AIME2025 (MathArena, 2025), and AIME2026 (MathArena, 2026). For Code, we evaluate on HumanEval+ and MBPP+ (Liu et al., 2023) and LiveCodeBench v5 and v6 (Jain et al., 2025). For Fact, we evaluate on HotpotQA, TriviaQA (Joshi et al., 2017), PopQA (Mallen et al., 2023), and 2WikiMultiHopQA (Ho et al., 2020), using 200 randomly sampled problems from each benchmark. We report pass@K across sampling budgets. More evaluation details are provided in Appendix B.

4.2 Main Results

Pass@K gains persist or widen from small to large K. OPD improves pass@1 in all eight pairs (Table 2), and in several settings these gains persist or even widen as the sampling budget increases.

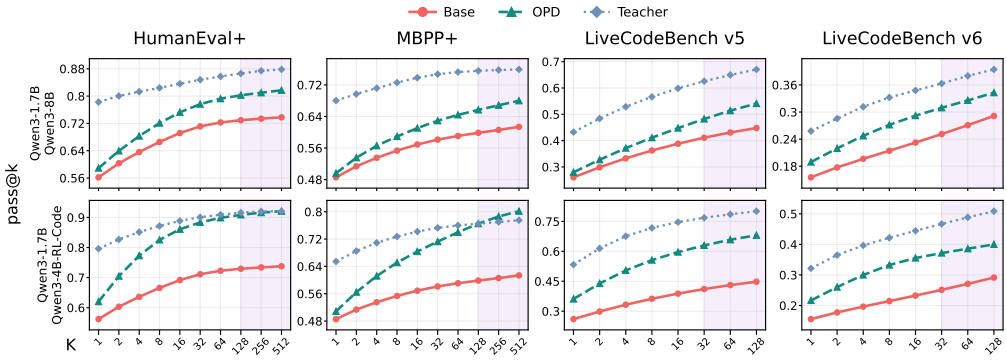


Figure 2: **Pass@K in two Code settings.** OPD gains persist and sometimes widen.

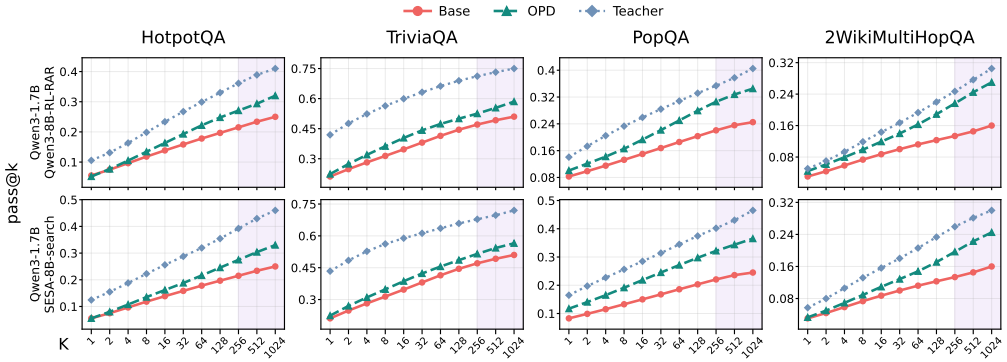


Figure 3: **Pass@K in the two Fact settings.** Both OPD-trained models outperform the pre-OPD base model on all four benchmarks at $K = 1024$, with no large-budget reversal.

In both Code settings, OPD stays ahead of the pre-OPD base model at every evaluated budget (Fig. 2). For Qwen3-1.7B ← Qwen3-4B-RL-Code (student ← teacher), the MBPP+ gain grows from 2.3 percentage points at $K = 1$ to 18.8 percentage points at $K = 512$. Similarly, with the same student and either Qwen3-8B-RL-RAR or SESA-8B-search as the teacher, both Fact settings remain ahead of the pre-OPD base model on all four benchmarks at $K = 1024$ (Fig. 3), with no reversal.

Pass@K gains may decrease or even reverse as K grows. In several Math settings, small-budget gains diminish as K increases and can eventually reverse. In the Qwen3-1.7B-Base ← Qwen3-4B-Base-GRPO setting and both DS-Distill-Qwen-1.5B settings using Skywork-OR1-Math-7B or JustRL-DS-1.5B as the teacher, several benchmark curves fall below the pre-OPD base model at large K (Fig. 4). For the Qwen3-1.7B-Base pair on AIME2024, OPD raises pass@1 by 7.8 percentage points but lowers pass@1024 by 16.7 points. Even for Qwen3-1.7B ← Qwen3-4B-RL-Math, where the average gain remains positive at the largest sampling budget, gains on some benchmarks diminish or reverse. This reflects the budget-dependent aggregation analyzed in Section 3.

Large-budget reversals also occur with other OPD methods. In this Qwen3 Math setting, all methods use Qwen3-1.7B-Base as the student and Qwen3-4B-Base-GRPO as the teacher; Direct-OPD also uses the pre-RL Qwen3-4B-Base checkpoint. We compare standard OPD with EOPD (Jin et al., 2026), which applies entropy-gated forward KL to regularize the student, and a pure forward KL variant without entropy gating. We also evaluate ExOPD (Yang et al., 2026b), which extrapolates teacher reward signal, and Direct-OPD (Feng et al., 2026), which uses the pre/post-RL teacher log-ratio as its training signal. Their training configurations are reported in Table 4. All five improve pass@1 on each of the four benchmarks, yet several fall below the pre-OPD base model at $K = 1024$ (Fig. 5), showing that large-budget degradation can also occur across alternative OPD formulations.

4.3 Why Can Pass@K Gains Persist or Reverse?

Accuracy-grouped contributions to dataset-level Δpass@K preserve their signs across K. As shown in the left panels of Fig. 6, we group problems by their pre-OPD accuracy and the direction

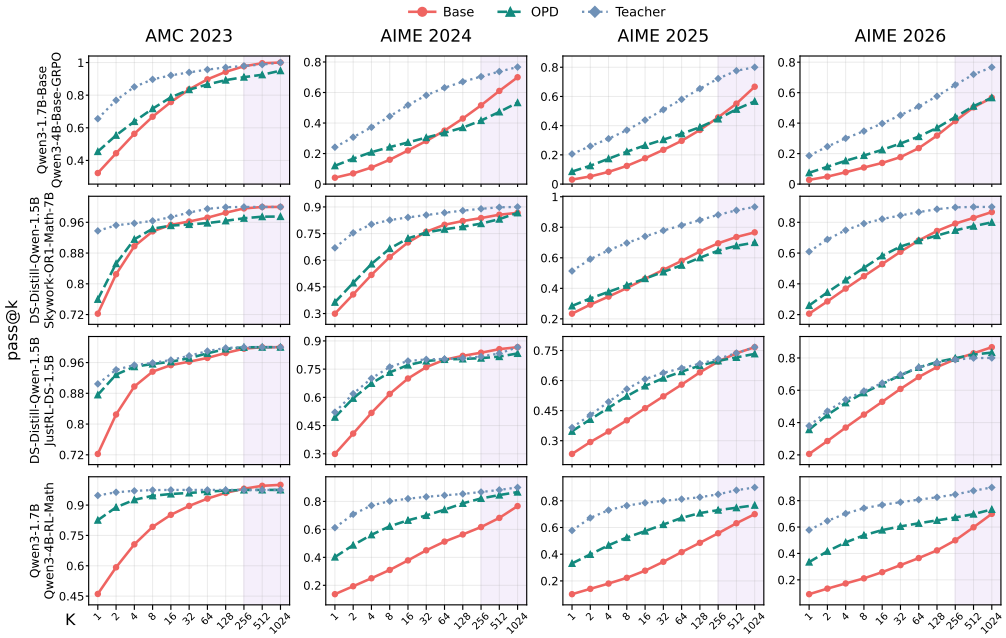


Figure 4: **Math pass@K.** OPD gains narrow and reverse on multiple benchmarks at large K.

of their accuracy change after OPD. Each group contributes either positively or negatively to the dataset-level Δpass@K, and the sign of its contribution remains unchanged as K increases, although its magnitude varies with K. This observation is consistent with our analysis in Lemma 1.

Aggregating these group contributions can lead to improvements or degradations in pass@K at large K. The right panels of Fig. 6 show their aggregate effect. On AIME2024 (Fig. 6a), positive contributions dominate at small K, but negative contributions grow to dominate at large K, resulting in a reversal. On PopQA (Fig. 6b), positive contributions remain dominant and the overall gain widens with K. These contrasting outcomes illustrate how the budget-dependent weighting in Section 3.2 shapes dataset-level pass@K changes across different sampling budgets and benchmarks.

5 Budget-Aware Pairwise Teacher Selection

We aim to select the teacher that leads to higher student pass@K at a target sampling budget. To this end, we introduce the *Teacher Advantage Score at K* (TAS@K), which ranks two candidate teachers for the same pre-OPD base model and training setup, at a chosen target budget K.

5.1 From Teacher Preference to a Budget-Aware Score

TAS@K computation. For each problem x in the target dataset $\mathcal{D}$, we sample N responses $\{y_i\}_{i=1}^{N}$ from the pre-OPD base model and estimate its accuracy $\widehat{p}_x = \frac{1}{N}\sum_{i=1}^{N} r_i$, where $r_i \in \{0,1\}$ indicates whether response y_i is correct. Given two candidate teachers A and B, we compute $d_i = \ell_A(y_i \mid x) - \ell_B(y_i \mid x)$, where ℓ_A and ℓ_B denote the mean token log likelihood of response y_i under teachers A and B, respectively. The TAS@K score is computed as follows:

$$\mathrm{TAS@}K(A,B) = \frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}} K\widehat{p}_x(1-\widehat{p}_x)^K \widehat{A}_x(A,B). \tag{7}$$

For problems with at least one correct and one incorrect sampled response, $\widehat{A}_x(A,B)$ is defined as:

$$\widehat{A}_x(A,B) = \frac{\sum_{i:r_i=1} d_i}{\sum_{i=1}^{N} r_i} - \frac{\sum_{i:r_i=0} d_i}{\sum_{i=1}^{N}(1-r_i)}. \tag{8}$$

If all sampled responses are correct or all are incorrect, we set $\widehat{A}_x(A,B) = 0$. With the definition, a positive TAS@K favors teacher A and a negative score favors teacher B for the target budget

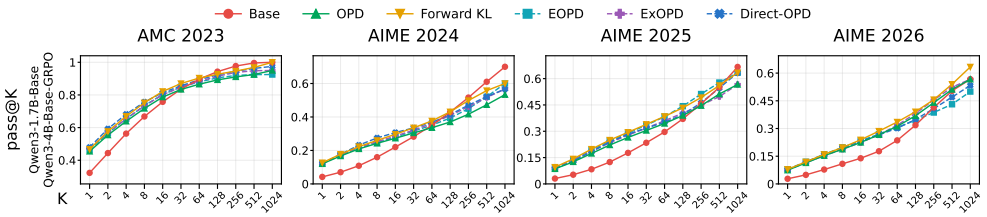


Figure 5: **Pass@K for other OPD methods.** Several show large-budget reversals.

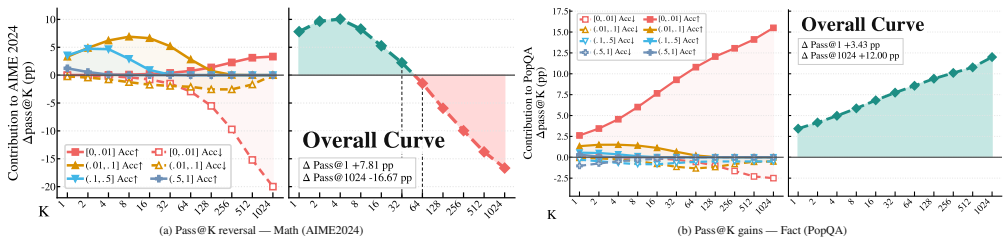


Figure 6: **Problem-level decomposition of pass@K changes.** Left panels group contributions by estimated pre-OPD accuracy and the direction of the accuracy change; right panels sum them, showing (a) a reversal on AIME2024 and (b) widening gains on PopQA.

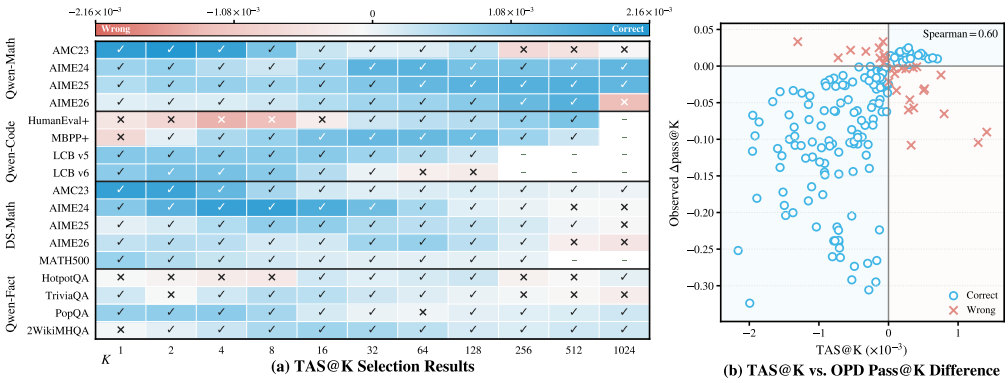


Figure 7: **Teacher selection with TAS@K.** Left: selection outcomes across settings and budgets. Blue/check and red/cross cells indicate correct and incorrect selections, respectively. Color intensity shows the absolute score in units of 10^{-3}; dashes mark unavailable budgets. Right: signed TAS@K versus the student pass@K difference between teachers A and B.

K. Notably, computing TAS@K does not involve any OPD-trained model; it only uses responses sampled from the pre-OPD base model and their likelihoods under the candidate teachers. Therefore, TAS@K can be computed before OPD training. Detailed derivations are provided in Appendix A.6.

5.2 Experimental Results of TAS@K

Settings. We evaluate four teacher pairs across Math, Code, and Fact: (i) Qwen-Math uses Qwen3-8B as teacher A and Qwen3-4B-RL-Math as teacher B; (ii) Qwen-Code uses Qwen3-8B as teacher A and Qwen3-4B-RL-Code as teacher B; (iii) DeepSeek-Math uses Skywork-OR1-Math-7B as teacher A and JustRL-DS-1.5B as teacher B; and (iv) Qwen-Fact uses Qwen3-8B-RL-RAR as teacher A and SESA-8B-search as teacher B. We use the benchmarks in Section 4.1, with MATH500 (Lightman et al., 2024) as additional benchmark included for DeepSeek-Math in that comparison.

TAS@K agrees with the observed student pass@K ordering in 83.6% of comparisons. The left panel of Fig. 7 shows that, across four teacher pairs and sampling budgets, TAS@K correctly identifies which teacher yields the higher OPD-trained pass@K in 83.6% of comparisons. This agreement is observed across Math, Code, and Fact settings, despite substantial variation in the teacher

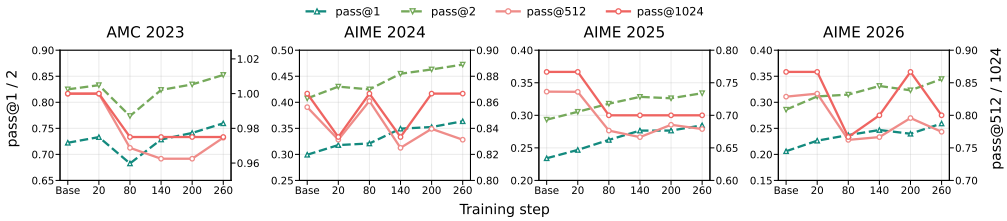


Figure 8: **Math training dynamics: small-budget gains, large-budget declines.** At Step 260, pass@1 exceeds Base on four benchmarks, but pass@1024 trails it on three.

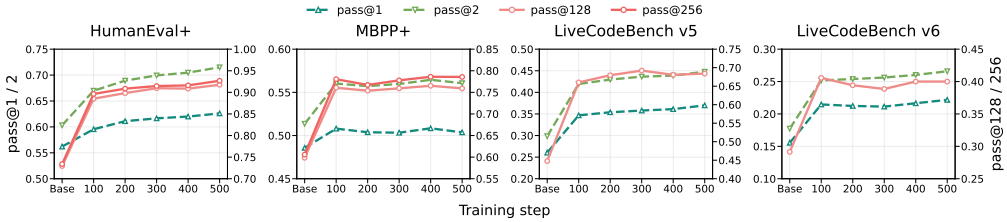


Figure 9: **Code training dynamics: gains appear early at small and large budgets.** From Step 100 onward, the pass@1 and largest-budget curves stay above Base in every panel.

pairs and benchmark characteristics. The preferred teacher can also change with K, indicating that teacher selection is budget-dependent and should account for the target sampling budget.

TAS@K exhibits a positive correlation with the observed pass@K difference. Beyond predicting the ordering, the signed TAS@K score correlates with the observed pass@K difference between the two OPD-trained students (Fig. 7, right), with Pearson and Spearman correlations of 0.46 and 0.60, respectively. The sign of TAS@K score generally matches the observed pass@K difference, while larger magnitudes demonstrate a tendency to be associated with larger performance gaps.

6 Further Analysis

Large-budget gains and reversals can emerge early during OPD training. As shown in Fig. 8 and Fig. 9, we compare intermediate checkpoints with the pre-OPD base model for DS-Distill-Qwen-1.5B ← Skywork-OR1-Math-7B (the Math pair) and Qwen3-1.7B ← Qwen3-4B-RL-Code (the Code pair). For the Math pair, pass@1 improves while pass@K shows an overall downward trend as K increases; by Step 80 on AIME2026, pass@1 is 3.1 percentage points above the pre-OPD base model while pass@1024 is 10.0 percentage points below it. For the Code pair, pass@1 improves together with an overall upward trend in pass@K; Step 100 already improves pass@1 and each benchmark's largest evaluated pass@K on all four code benchmarks. These results show that the large-K behavior observed after OPD training can already emerge at these early stages of intermediate training steps.

OPD trajectories are favored by the teacher while remaining likely under the pre-OPD base model. For the Math pair, Fig. 10 compares the perplexity of trajectories sampled from the pre-OPD base, OPD-trained, and teacher models, scored by both the pre-OPD base model and the teacher. OPD-generated trajectories receive lower perplexity than pre-OPD base trajectories under the teacher scorer, while remaining even more likely than both pre-OPD base and teacher trajectories under the pre-OPD base scorer. This suggests that OPD shifts generation toward trajectories better aligned with the teacher while remaining within high-likelihood regions of the pre-OPD base distribution. More details on the sampling and scoring setup are in Appendix F.

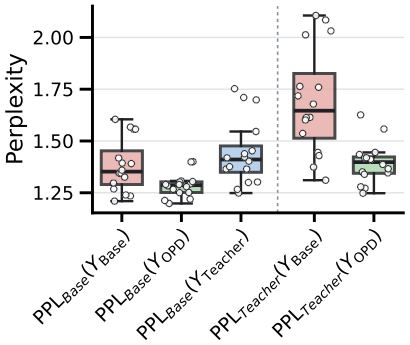


Figure 10: **Math trajectory perplexity.** Y_{Base}, Y_{OPD}, and Y_{Teacher} denote trajectory sources; PPL_{Base} and $\text{PPL}_{\text{Teacher}}$ denote the respective scoring models.

An illustrative case study. Fig. 11 compares two correct responses to AMC2023 12A Problem 4, where the OPD-trained model is substantially more

AMC2023 12A, Problem 4: How many digits are in the base-ten representation of $8^5 \cdot 5^{10} \cdot 15^5$?

Base trajectory (221/1024). $8 = 2^3$, $15 = 3 \cdot 5$, so $N = 8^5\,5^{10}\,15^5 = 2^{15}\,3^5\,5^{15}$. Then $\log_{10} N = 15\log_{10} 2 + 5\log_{10} 3 + 15\log_{10} 5 \approx 17.3855$, so N has $\lfloor 17.3855 \rfloor + 1 = 18$ digits: $\boxed{18}$.

Reading note. Completes the factorization and digit-count route.

OPD trajectory (893/1024). Step 1: $8^5\,5^{10}\,15^5 = 2^{15}\,3^5\,5^{15}$. **Step 2:** $2^{15}\,3^5\,5^{15} = 10^{15} \cdot 243$. **Step 3:** 10^{15} appends 15 zeros to 243, so $15 + 3 = 18$ digits: $\boxed{18}$.

Reading note. Follows the same route, then uses $10^{15} \cdot 243$.

Figure 11: **Case study of a problem both models solve.** At $K = 1024$, the pre-OPD base model achieves an accuracy of 221/1,024, compared with 893/1,024 for the OPD-trained model. Both models generate similar reasoning patterns, such as the same factorization route.

accurate. Both use the same factorization but finish differently: the pre-OPD base trajectory uses logarithms, whereas the OPD trajectory rewrites it with a power of ten. This shows that both models can share the same solution structure while taking different, equally correct trajectories.

7 Related Work

On-policy distillation. Classical and sequence-level distillation transfer teacher predictions or generated sequences (Hinton et al., 2015; Kim & Rush, 2016); GKD (Agarwal et al., 2024) and MiniLLM (Gu et al., 2024) instead use student-generated trajectories for teacher feedback. Recent studies identify prefix mismatch and estimator bias (Fu et al., 2026a; Zhu et al., 2026), while others examine early parameter updates and training efficiency (Cai et al., 2026). REOPOLD (Ko et al., 2026), entropy-aware forward KL (Jin et al., 2026), and trust-region OPD (Xing et al., 2026) seek more stable supervision or better preservation of diversity; Direct-OPD (Feng et al., 2026) transfers RL-induced policy shifts, whereas ExOPD (Yang et al., 2026b) extrapolates them. Alongside these studies of training signals and objectives, we examine how the resulting student's pass@K changes as the test-time sampling budget increases, including whether gains persist or reverse.

Sampling efficiency and coverage. Studies of RLVR report low-/high-budget crossovers and persistent gains (Yue et al., 2025; Wu et al., 2025; Liu et al., 2025); MATH-Beyond examines expansion beyond the base model's finite-budget solved set (Mayilvahanan et al., 2026). For OPD, one study reports gains converging at large K and examines the quality of teacher feedback (Wang et al., 2026). Sampled-demonstration self-distillation can flatten pass@K curves (Nicolicioiu et al., 2026). An OPD pass@1/pass@16 reversal on MBPP+ appears in the evaluation of influence-directed distillation (Yang et al., 2026a). We examine both reversals and persistent gains across Math, Code, and Fact. We explain how problem-level accuracy changes receive different weights as K grows and, through an idealized analysis of OPD's full-vocabulary reverse KL update, show why some problems can worsen during finite training even when the teacher is more accurate on every problem.

Teacher suitability and selection. Teacher-student compatibility matters for OPD (Li et al., 2026). RSR (Yang et al., 2026c) assesses the suitability of teacher-generated trajectories for supervised fine-tuning and uses this signal for teacher selection, whereas TrustMOPD (Sun et al., 2026) allocates supervision among teachers at each student-generated prefix. We instead use a score computed before OPD training to select the teacher expected to yield higher student pass@K at a target budget, given two candidates for the same pre-OPD base model under the same training setup.

8 Conclusion

In this work, we study on-policy distillation through the lens of test-time scaling. Across the evaluated settings, we observe two distinct patterns: pass@K gains can persist or widen from small to large sampling budgets, or improve at small budgets but reverse at large budgets. Our theoretical analysis derives a sufficient condition for such large-budget degradation based on how problem-level accuracy changes contribute to pass@K across sampling budgets. Additionally, we introduce TAS@K to guide teacher selection at a target sampling budget, with its rankings matching the observed student pass@K ordering in 83.6% of comparisons. Overall, we hope these findings encourage future OPD studies to account for test-time sampling budgets in both evaluation and teacher selection in practice.

AI USE STATEMENT

We used generative AI tools to improve the clarity and presentation of the manuscript, search for and identify relevant literature, support research ideation and execution, and assist in formulating mathematical claims. The authors reviewed the AI-assisted content incorporated into this work. In particular, we checked retrieved references against their original sources, assessed research suggestions before adopting them, and checked the mathematical arguments for correctness. The authors retain full responsibility for the accuracy, originality, and integrity of the manuscript and associated research artifacts, and any remaining errors or omissions are our own.

ETHICS STATEMENT

This work studies on-policy distillation using established benchmarks in mathematics, code generation, and factual question answering. It does not involve human participant studies or introduce a newly collected dataset. The proposed teacher-selection score may help improve model capabilities, including code generation, and could have downstream misuse risks. A gain in pass@K at one sampling budget should not be assumed to hold at other budgets or interpreted as evidence of safety or reliability in deployment. Large-budget sampling also entails computational and environmental costs. We report results across sampling budgets and describe the evaluation conditions to support an appropriately scoped interpretation of these findings and to make their limitations explicit.

REPRODUCIBILITY STATEMENT

We take several steps to support reproducibility. Section 3 and Appendix A provide the assumptions and proofs for our theoretical results. Section 4.1 and Appendices B and C document the benchmark and checkpoint sources, training and evaluation configurations, prompts, decoding settings, and scoring rules. We also provide an anonymous code package at https://anonymous.4open.science/r/review-artifact-4B60/, containing a compact subset of the training and evaluation code used in our experiments, runnable examples with synthetic data, and a TAS@K calculation. Together, these materials allow readers to inspect the theoretical arguments and the main experimental procedures, and to run the included examples and the TAS@K calculation.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024. URL https://arxiv.org/abs/2407.21787v1.

Yuchen Cai, Ding Cao, Liang Lin, Chunxi Luo, Xin Xu, Kai Yang, Weijie Liu, Saiyong Yang, Tianxiang Zhao, Guangzhong Sun, Guiquan Liu, and Junfeng Fang. Learning to foresee: Unveiling the unlocking efficiency of on-policy distillation. arXiv preprint arXiv:2605.11739, 2026. URL https://arxiv.org/abs/2605.11739v3.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating

large language models trained on code. arXiv preprint arXiv:2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

Tong Chen, Akari Asai, Luke Zettlemoyer, Hannaneh Hajishirzi, and Faeze Brahman. Train for truth, keep the skills: Binary retrieval-augmented reward mitigates hallucinations. *arXiv preprint arXiv:2510.17733*, 2025.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026. doi: 10.48550/arXiv.2606.19348. URL https://arxiv.org/abs/2606.19348.

Shiyuan Feng, Huan-ang Gao, Haohan Chi, Hanlin Wu, Zhilong Zhang, Zheng Jiang, Bingxiang He, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Weak-to-strong generalization via direct on-policy distillation. arXiv preprint arXiv:2607.05394, 2026. URL https://arxiv.org/abs/2607.05394.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. arXiv preprint arXiv:2603.25562, 2026a. URL https://arxiv.org/abs/2603.25562v2.

Zenghuang Fu, Zhaoyang Li, Qiuyuan Ai, Haoyu Wu, Minghui Wu, Chenxu Zhao, Ante Wang, Guannan He, and Changwei Wang. Self-play meets skill evolution: Self-evolving search agents that pose, solve, and remember. *arXiv preprint arXiv:2607.29468*, 2026b. URL https://arxiv.org/abs/2607.29468.

ganler. CodeR1-12K. Hugging Face dataset repository, 2025. URL https://huggingface.co/datasets/ganler/code-r1-12k. Frozen revision 2a0d26ddd96adc28138ad9cd2c371c27ba5614ac.

Xinmu Ge. math-eval: Reproducible mathematical reasoning generation and evaluation, July 2026a. URL https://doi.org/10.5281/zenodo.21411208. Software, Apache-2.0 license.

Xinmu Ge. math-vault: Curated, traceable snapshots of public mathematical reasoning datasets, 2026b. URL https://doi.org/10.5281/zenodo.21411214. Dataset, version v0.1.0.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=5h0qf7IBZZ.

Daya Guo, Dejian Yang, Haowei Zhang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

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. arXiv preprint arXiv:2512.16649, 2025a. URL https://arxiv.org/abs/2512.16649.

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. Skywork open reasoner 1 technical report. arXiv preprint arXiv:2505.22312, 2025b. URL https://arxiv.org/abs/2505.22312.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015. URL https://arxiv.org/abs/1503.02531.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020.

HuggingFaceH4. AIME 2024. Hugging Face dataset repository, 2025. URL https://huggingface.co/datasets/HuggingFaceH4/aime_2024. AIME I and II; frozen revision 2fe88a2f1091d5048c0f36abc874fb997b3dd99a.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pp. 58791–58831, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/94074dd5a072d28ff75a76dabed43767-Abstract-Conference.html.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=J5i09faOOf.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

Jongwoo Ko, Sara Abdali, Young Jin Kim, Tianyi Chen, and Pashmina Cameron. Scaling reasoning efficiently via relaxed on-policy distillation. arXiv preprint arXiv:2603.11137, 2026. URL https://arxiv.org/abs/2603.11137v1.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

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. In *ICML 2026 Workshop on Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning*, 2026. URL https://arxiv.org/abs/2604.13016. Non-archival workshop paper.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *ICLR*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/aca97732e30bcf1303bc22ac3924fd16-Abstract-Conference.html.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572, 2023.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. ProRL: Prolonged reinforcement learning expands reasoning boundaries in large language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/1a22b912945fb7c0bdd079e792b31b6f-Abstract-Conference.html.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 9802–9822, 2023.

math-ai. math-ai/amc23 dataset. Hugging Face dataset repository, 2025. URL https://huggingface.co/datasets/math-ai/amc23. Frozen revision 80815d37005feb82cd7f8fbc6901d5d3eff43057.

MathArena. AIME 2025. Hugging Face dataset repository, 2025. URL https://huggingface.co/datasets/MathArena/aime_2025. AIME I and II; frozen revision c94da77eb22bbd6439e62a323bec18493a421302.

MathArena. AIME 2026. Hugging Face dataset repository, 2026. URL https://huggingface.co/datasets/MathArena/aime_2026. AIME I and II; frozen revision d2de22f3c656b4f56cf8981212186377d1e23bc3.

Prasanna Mayilvahanan, Ricardo Dominguez-Olmedo, Thaddäus Wiedemer, and Wieland Brendel. MATH-Beyond: A benchmark for RL to expand beyond the base model. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RNkErKpCAp.

Andrei Liviu Nicolicioiu, Mohammad Pezeshki, and Aaron Courville. On-policy self-distillation with sampled demonstrations reduces output diversity, 2026. URL https://arxiv.org/abs/2606.26091.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026.

Jie Sun, Mao Zheng, Mingyang Song, Zeyuan Liu, Gengsheng Li, Houcheng Jiang, Yilin Cheng, Bichuan Feng, Yuchen Cai, Junfeng Fang, and Xiang Wang. Distill what you trust: Reliability-aware multi-teacher on-policy distillation. arXiv preprint arXiv:2609.23697, 2026. URL https://arxiv.org/abs/2609.23697v1.

Rui Wang, Hongru Wang, Yi Chen, Boyang Xue, Tianqing Fang, Wenhao Yu, and Kam-Fai Wong. Demystifying on-policy distillation: Roles, pathologies, and regulations. arXiv preprint arXiv:2607.13399, 2026. URL https://arxiv.org/abs/2607.13399.

Fang Wu, Weihao Xuan, Ximing Lu, Mingjie Liu, Yi Dong, Zaid Harchaoui, and Yejin Choi. The invisible leash? why RLVR may or may not escape its origin. arXiv preprint arXiv:2507.14843, 2025. URL https://arxiv.org/abs/2507.14843v4.

Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation. arXiv preprint arXiv:2606.01249, 2026. URL https://arxiv.org/abs/2606.01249.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. doi: 10.48550/arXiv.2505.09388. URL https://arxiv.org/abs/2505.09388v1.

Run Yang, Runpeng Dai, Jie Sun, Jielei Zhang, Fan Zhou, Hongtu Zhu, Peiyi Li, and Longwen Gao. Influence-directed distillation: Solving the diversity bottleneck in sampled-token on-policy distillation. arXiv preprint arXiv:2608.29846, 2026a. URL https://arxiv.org/abs/2608.29846v1.

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

Yuming Yang, Mingyoung Lai, Wanxu Zhao, Xiaoran Fan, Zhiheng Xi, Mingqi Wu, Chiyue Huang, Jun Zhao, Haijun Lv, Jian Tong, Yunhua Zhou, Yicheng Zou, Qipeng Guo, Tao Gui, Qi Zhang, and Xuanjing Huang. Which reasoning trajectories teach students to reason better? a simple metric of informative alignment. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 42123–42150, San Diego, California, United States, July 2026c. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.1950. URL https://aclanthology.org/2026.acl-long.1950/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

Zhuolin Yang, Zihan Liu, Yang Chen, Wenliang Dai, Boxin Wang, Sheng-Chieh Lin, Chankyu Lee, Yangyi Chen, Dongfu Jiang, Jiafan He, Renjie Pi, Grace Lam, Nayeon Lee, Alexander Bukharin, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nemotron-Cascade 2: Post-training LLMs with cascade RL and multi-domain on-policy distillation. *arXiv preprint arXiv:2603.19220*, 2026d. doi: 10.48550/arXiv.2603.19220. URL https://arxiv.org/abs/2603.19220v2.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, 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, 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. arXiv preprint arXiv:2503.14476, 2025. URL https://arxiv.org/abs/2503.14476v2.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In Danielle Belgrave, Cheng Zhang, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, and Nancy Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 57654–57689. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/537d5aa768c2d534016a4d06f87bc8fb-Paper-Conference.pdf.

Siqi Zhu, Xuyan Ye, Hongyu Lu, Weiye Shi, and Ge Liu. The many faces of on-policy distillation: Pitfalls, mechanisms, and fixes, 2026. URL https://arxiv.org/abs/2605.11182.

A Proofs and Scope of the Analysis

A.1 Problem-Level Accuracy and Dataset-Level Changes

Following the main text, the problem-level metric is

$$\text{pass@}K(x;\theta) = 1 - (1 - p_\theta(x))^K,$$

and the dataset-level metric is $\text{pass@}K(\theta) = \mathbb{E}_x[\text{pass@}K(x;\theta)]$. Conditional independence across the K samples gives the following problem-level decomposition of the dataset-level changes

$$\Delta\text{pass@}1 = \mathbb{E}_x[\Delta p(x)], \qquad \Delta\text{pass@}K = \mathbb{E}_x[\Delta\text{pass@}K(x)].$$

Throughout, Δ denotes the post-training value minus the pre-training value. The expression $1 - (1 - p)^K$ assumes that the K responses are independently sampled from the same model distribution.

The exact finite change is

$$\Delta\text{pass@}K(x) = (1 - p_{\theta_0}(x))^K - (1 - p_{\theta_T}(x))^K = \int_{p_{\theta_0}(x)}^{p_{\theta_T}(x)} K(1 - r)^{K-1}\, dr.$$

Proof of Lemma 1. The function $1 - (1 - p)^K$ is strictly increasing on $[0, 1]$; for $K > 1$, its derivative vanishes at $p = 1$, which does not affect strict monotonicity. Its integral form implies that $\Delta\text{pass@}K(x)$ has the same sign as $\Delta p(x)$. For fixed K, Taylor expansion at $p_{\theta_0}(x)$ yields

$$\Delta\text{pass@}K(x) = K(1 - p_{\theta_0}(x))^{K-1}\Delta p(x) + O(\Delta p(x)^2).$$

□

For $K > 1$, local sensitivity to an absolute accuracy change is greater at lower initial accuracy. For the same problem, the direction of the pass@K change is consistent across sampling budgets.

Proof of Proposition 1. Let $G = \mathbb{E}_x \max\{\Delta p(x), 0\}$ and $L = \mathbb{E}_x \max\{-\Delta p(x), 0\} > 0$. The left inequality in equation 4 gives $\Delta\text{pass@}1 = G - L > 0$. For gain problems, the integral kernel is at most $K(1 - p_+)^{K-1}$; for loss problems, it is at least $K(1 - p_-)^{K-1}$. Therefore,

$$\Delta\text{pass@}K \le K(1 - p_+)^{K-1}G - K(1 - p_-)^{K-1}L < 0. \tag{9}$$

The final strict inequality is exactly the ratio condition in the proposition. Its right-hand side is a conservative lower bound on relative sensitivity, so the condition is sufficient but not necessary. □

For example, under the proposition's probability-region separation assumption, let $K = 10$, $p_+ = .2$, $p_- = .02$, and $G = 3L$. The sensitivity bounds for gains and losses are approximately 1.342 and 8.337, respectively; the loss-to-gain ratio is approximately $6.212 > G/L = 3$. Thus, $\Delta\text{pass@}1 = 2L > 0$, whereas the loss bound for the same example at the larger budget $K = 10$ gives

$$\Delta\text{pass@}10 \le [30(.8)^9 - 10(.98)^9]L \simeq -4.311L < 0.$$

If $p_{\theta_\tau}(x)$ is absolutely continuous along a finite training path and $\int_0^T \mathbb{E}_x[|b_x(\tau)|]d\tau < \infty$, the chain rule and interchange of integration then give the following integral expression.

$$\Delta\text{pass@}K = \int_0^T \mathbb{E}_x\left[K(1 - p_{\theta_\tau}(x))^{K-1}b_x(\tau)\right] d\tau, \tag{10}$$

A.2 OPD Updates and Prefix Weighting

Let $x' \sim \mathcal{D}$ and $y \sim \pi_{\theta_{\text{old}}}(\cdot \mid x')$ denote a student rollout drawn with fixed sampling parameters θ_{old}, and let $s = (x', y_{<t})$. In this subsection, L denotes trajectory length, distinct from the preceding loss quantity, and we assume $0 < \mathbb{E}[L] < \infty$. The effective token distribution induced by the trajectory-sum objective satisfies

$$\mathbb{E}_{s\sim\rho_{\theta_{\text{old}}}}[g(s)] = \frac{\mathbb{E}_{x',y}[\sum_{t=1}^{L} g(x', y_{<t})]}{\mathbb{E}_{x',y}[L]}.$$

The normalized local loss is therefore

$$\mathcal{L}(\theta;\theta_{\text{old}}) = \mathbb{E}_{s\sim\rho_{\theta_{\text{old}}}}[D(\theta;s)].$$

It equals the trajectory-sum loss under fixed rollouts divided by $\mathbb{E}[L]$; matching one update between the two objectives requires $\eta_{\text{normalized}} = \mathbb{E}[L]\eta_{\text{sum}}$. Different token or microbatch weights require a corresponding redefinition of ρ and of the length normalization $\mathbb{E}[L]$ that it induces.

We differentiate $\mathcal{L}$ with respect to its first argument and stop gradients through the rollout. The continuously resampled limit is

$$\dot{\theta}_\tau = -\mathbb{E}_{s\sim\rho_{\theta_\tau}}[\nabla_\theta D(\theta_\tau;s)], \qquad b_x(s,\tau) := \langle \nabla_\theta p_{\theta_\tau}(x), -\nabla_\theta D(\theta_\tau;s)\rangle.$$

Here, $b_x(s,\tau)$ is a single-prefix contribution, whereas $b_x(\tau) = \mathbb{E}_{s\sim\rho_{\theta_\tau}}[b_x(s,\tau)]$ is the problem-level average rate. Under differentiability, integrable domination, and interchangeability of differentiation and expectation, $\frac{d}{d\tau}p_{\theta_\tau}(x) = b_x(\tau)$. This is neither the full gradient that differentiates through the on-policy sampling distribution nor the exact dynamics of Adam, finite batches, or top-k distillation. The first-order response of an actual discrete update must use its realized parameter displacement; when top-p or top-k changes the support, global smoothness cannot be assumed across a switching point. The continuous progress variable τ can absorb proportional rescalings of loss normalization and step size; it is not directly an optimization step count or measured wall-clock time.

A.3 Accuracy Dynamics under Full-Vocabulary Reverse KL

We first consider the full-vocabulary idealization $D(\theta;s) = D_{\text{KL}}(\pi_\theta(\cdot \mid s)\|\pi_T(\cdot \mid s))$ with full support for both distributions. This is a sufficient setting for the following calculation, not the top-k token KL used in the main experiments:

$$\nabla_\theta D(\theta;s) = \sum_a \pi_\theta(a \mid s)\log\frac{\pi_\theta(a \mid s)}{\pi_T(a \mid s)}\nabla_\theta \log \pi_\theta(a \mid s),$$

where the $+1$ term vanishes by the zero-mean score identity. Define the gradient alignment

$$h_{x,a}(s,\tau) = \langle \nabla_\theta p_{\theta_\tau}(x), \nabla_\theta \log \pi_{\theta_\tau}(a \mid s)\rangle.$$

The contribution of the individual prefix state s to the problem-level average rate is then

$$b_x(s,\tau) = \sum_a \pi_{\theta_\tau}(a \mid s)\log\frac{\pi_T(a \mid s)}{\pi_{\theta_\tau}(a \mid s)}h_{x,a}(s,\tau).$$

Thus, reverse KL divergence alone does not determine the sign of b_x.

For the full-response distribution induced by a fixed evaluation decoding rule, let $R_x(y) \in \{0,1\}$ denote correctness and assume $p_{\theta_\tau}(x) \in (0,1)$ along the path. All conditional expectations below are taken under $y \sim \pi_{\theta_\tau}(\cdot \mid x)$. Define

$$a_x(\tau) = \left\langle \mathbb{E}[\nabla_\theta \log \pi_{\theta_\tau}(y \mid x) \mid R_x = 1] - \mathbb{E}[\nabla_\theta \log \pi_{\theta_\tau}(y \mid x) \mid R_x = 0], \dot{\theta}_\tau \right\rangle.$$

The score identity gives $\frac{d}{d\tau}p_{\theta_\tau}(x) = p_{\theta_\tau}(x)(1-p_{\theta_\tau}(x))a_x(\tau)$. If a_x is integrable, then

$$\frac{d}{d\tau}\operatorname{logit} p_{\theta_\tau}(x) = a_x(\tau), \qquad p_{\theta_T}(x) = \sigma\left(\operatorname{logit} p_{\theta_0}(x) + \int_0^T a_x(\tau)d\tau\right).$$

Here, $\operatorname{logit} p = \log[p/(1-p)]$ and $\sigma(z) = 1/(1+e^{-z})$. This identity applies to any differentiable evaluation-policy path satisfying these conditions; it is not specific to full-vocabulary KL. The evaluation decoding rule may differ from the training sampling rule, but both p and the scores above must use the same evaluation distribution. Full-vocabulary KL only specifies the update direction above, not the magnitude or sign of the induced change in problem-level accuracy.

A.4 A TWO-PROBLEM COUNTEREXAMPLE

Proof of Proposition 2. For $K = 1$, the initial dataset-level rate is $\mathbb{E}_x[b_x(0)]$. For any fixed $K > 1$, the chain rule gives the following expression for the initial rate of the dataset-level metric

$$\frac{d}{d\tau}\text{pass@}K(\theta_\tau)\Big|_{\tau=0} = \mathbb{E}_x\left[K(1-p_{\theta_0}(x))^{K-1}b_x(0)\right].$$

The two strict inequalities in equation 6, differentiability, and a first-order expansion therefore give the claimed signs for every sufficiently small positive finite update.

It remains to show that the criterion can hold despite a teacher that is more accurate on every problem. Consider two equally weighted, independent three-class softmax problems, with the first class correct:

$$P_1 = (.3, .35, .35),\ Q_1 = (.5, .25, .25), \qquad P_2 = (.001, .899, .1),\ Q_2 = (.002, .997, .001).$$

The teacher assigns a higher probability to the correct class on both problems. Apply standard gradient descent to the average KL loss $(D_1 + D_2)/2$, where $D_j = D_{\mathrm{KL}}(P_j \| Q_j)$. For problem $j \in \{1, 2\}$ and class $i \in \{1, 2, 3\}$, let $r_{ji} = \log(Q_{ji}/P_{ji})$ and $\bar{r}_j = \sum_i P_{ji} r_{ji}$. The negative logit gradient of the single-problem KL is $v_{ji} = P_{ji}(r_{ji} - \bar{r}_j)$; hence, the actual direction for the average loss is $v_{ji}/2$, and

$$b_j = \frac{P_{j1}}{2}\left(v_{j1} - \sum_i P_{ji} v_{ji}\right).$$

Substitution gives

$$b_1 \simeq .02802438 > 0, \qquad b_2 \simeq -.00016832 < 0.$$

Therefore,

$$\frac{d}{d\tau}\text{pass@}1(\theta_\tau)\Big|_{\tau=0} = (b_1 + b_2)/2 \simeq .01392803 > 0,$$

$$\frac{d}{d\tau}\text{pass@}1024(\theta_\tau)\Big|_{\tau=0} = 512[(.7)^{1023}b_1 + (.999)^{1023}b_2] \simeq -.03096647 < 0.$$

For example, the expressions above verify $0.02802 < b_1 < 0.02803$ and $-0.0001684 < b_2 < -0.0001683$; together with $0.7^{1023} < 10^{-158}$ and $0.999^{1023} > 1/3$, these bounds ensure that the two derivatives have strictly opposite signs. Smoothness further gives

$$\Delta_h \text{pass@}K = h\ \frac{d}{d\tau}\text{pass@}K(\theta_\tau)\Big|_{\tau=0} + O(h^2).$$

Hence, for sufficiently small $h > 0$, the changes have different directions at different sampling budgets. With step size $h = .2$ for the average loss (equivalently, .1 for the summed KL), updating the logits and reapplying softmax gives

$$p_1 : .3 \to .30563462, \quad p_2 : .001 \to .00096659,$$

$$\text{pass@}1 : .1505 \to .15330060, \qquad \text{pass@}1024 : .82051426 \to .81426152,$$

and $D_1 : .08228288 \to .07758400$ and $D_2 : .36680638 \to .33247070$; thus, both problem-level KL divergences decrease over this finite step, as expected under a reverse-KL update.

Now replace the teacher for the second problem with

$$Q_2' = (.002, .998{\cdot}.899/.999, .998{\cdot}.1/.999).$$

This preserves the teacher's accuracy on both problems, while making the teacher-to-student probability ratios identical across the incorrect classes. Let $p = P_{21} = .001$ and $q = Q_{21}' = .002$. With $c = \operatorname{logit} q - \operatorname{logit} p > 0$, direct simplification gives

$$b_2' = \frac{p^2 c}{2}\left[(1-p)^2 + P_{22}^2 + P_{23}^2\right] \simeq 6.3035711 \times 10^{-7} > 0.$$

Thus, the same student and identical problem-level teacher accuracies can produce opposite signs of the accuracy change on the second problem. Accuracy vectors alone are insufficient to determine this sign under a full-vocabulary reverse KL update. □

This softmax construction establishes a possibility under reverse KL divergence. It is not empirical evidence from LLMs, nor does it establish that the behavior is unique to this divergence.

A.5 Scope of the Finite-Training Analysis

If $p_{\theta_\tau}(x)$ is absolutely continuous, then $p_{\theta_T}(x) - p_{\theta_0}(x) = \int_0^T b_x(\tau)d\tau$. This describes only a finite path: local positive or negative rates cannot be extrapolated to a convergence claim. If, at the end of training, $\pi_{\theta_T}(\cdot \mid x) = \pi_T(\cdot \mid x)$ holds on every evaluation problem under the same decoding rule, and the teacher is no worse than the pre-OPD student on every problem, then $p_{\theta_T}(x) \geq p_{\theta_0}(x)$, so neither problem-level nor dataset-level pass@K degrades for any K. Lowering KL only on training prefix states does not ensure this evaluation-distribution match; teacher realizability, state coverage, and optimization convergence are likewise not implied by the local analysis above.

A.6 Derivation of TAS@K

Fix a problem x and suppress its subscript. Let S be the pre-OPD base-model response distribution, T a teacher distribution, and $r(y) \in \{0, 1\}$ the correctness indicator. Assume $p, q \in (0, 1)$, $D_{\mathrm{KL}}(s^+ \| t^+) < \infty$, $D_{\mathrm{KL}}(s^- \| t^-) < \infty$, and that the displayed log-ratio expectations are finite. Write

$$S = p\, s^+ + (1-p)s^-, \qquad T = q\, t^+ + (1-q)t^-,$$

where s^+ and s^- are the base-model distributions conditional on correct and incorrect responses, respectively, and t^+, t^- are the analogous teacher conditionals. Consider the restricted student family

$$\pi_z = z\, s^+ + (1-z)s^-,$$

which changes only the problem-level accuracy while preserving the two within-group distributions. This restriction is an analytic model, not an assumption that a real LLM update leaves within-group composition unchanged; it isolates the effect of problem-level accuracy alone in the analysis.

Let $d^+ = D_{\mathrm{KL}}(s^+ \| t^+)$ and $d^- = D_{\mathrm{KL}}(s^- \| t^-)$. Because the supports of s^+ and s^- are disjoint,

$$D_{\mathrm{KL}}(\pi_z \| T) = D_{\mathrm{KL}}(\mathrm{Bern}(z) \| \mathrm{Bern}(q)) + z d^+ + (1-z)d^-. \tag{11}$$

Let z^* denote the student accuracy that minimizes $D_{\mathrm{KL}}(\pi_z \| T)$. Differentiating in z yields

$$\operatorname{logit} z^* = \operatorname{logit} q - (d^+ - d^-).$$

Define the teacher's sequence-level likelihood advantage over the pre-OPD base model as

$$a_T(y) = \log T(y \mid x) - \log S(y \mid x),$$

and its correct-versus-incorrect preference gap as

$$M_x(T) = \mathbb{E}_{s^+}[a_T(y)] - \mathbb{E}_{s^-}[a_T(y)].$$

Direct substitution gives

$$M_x(T) = \operatorname{logit} q - \operatorname{logit} p - (d^+ - d^-),$$

and hence, at the restricted optimum, the student accuracy satisfies, on the logit scale,

$$\operatorname{logit} z_x^*(T) = \operatorname{logit} p_x + M_x(T). \tag{12}$$

For two teachers A and B, define their sequence-level pairwise advantage as

$$A_x(A, B) = \mathbb{E}_{y \sim s_x^+}\left[\log \frac{T_A(y \mid x)}{T_B(y \mid x)}\right] - \mathbb{E}_{y \sim s_x^-}\left[\log \frac{T_A(y \mid x)}{T_B(y \mid x)}\right]. \tag{13}$$

Subtracting Eq. (12) for the two teachers then gives

$$\operatorname{logit} z_{A,x}^* - \operatorname{logit} z_{B,x}^* = M_x(T_A) - M_x(T_B) \equiv A_x(A, B). \tag{14}$$

Here, logit is the logarithm of the ratio of correct to incorrect response probability. The identity states that the difference between the two optimal student accuracies on this scale equals the teachers' relative preference gap. Since logit is strictly increasing, the teacher whose sequence-level preferences more strongly favor correct responses yields the higher optimal accuracy on this problem. This conclusion assumes exact reverse KL minimization within the restricted student family.

To motivate the operational score, suppose both restricted optima remain near the base accuracy p_x and their pairwise preference difference is small. Expanding the inverse logit around p_x gives

$$z^*_{A,x} - z^*_{B,x} \approx p_x(1-p_x)A_x(A,B).$$

Combining this with the pass@K derivative $K(1-p_x)^{K-1}$ gives the first-order contribution

$$Kp_x(1-p_x)^K A_x(A,B),$$

which motivates the budget weight used by TAS@K in Eq. (7) at a target budget K.

Empirically, both teachers score the same responses $y_i \sim S$. With sequence-sum log likelihood and the same within-group weights, the $-\log S(y_i \mid x)$ term cancels teacher by teacher, yielding an estimator of $A_x(A,B)$. Our operational score instead uses the length-normalized mean token log likelihood of each response when scoring the two teachers, rather than the sequence-sum form

$$\bar{\ell}_{T,i} = \frac{1}{L_i}\sum_{t=1}^{L_i} \log T(y_{i,t} \mid x, y_{i,<t}), \qquad d_i = \bar{\ell}_{T_A,i} - \bar{\ell}_{T_B,i}.$$

For correctness labels $r_i \in \{0,1\}$, the empirical problem-level advantage is

$$\widehat{A}_x(A,B) = \begin{cases} \mathbb{E}[d_i \mid r_i = 1] - \mathbb{E}[d_i \mid r_i = 0], & 0 < \sum_i r_i < n_x, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Thus, the operational $\widehat{A}_x$ is a length-normalized empirical surrogate rather than the exact sequence-level quantity in Eq. (13). Problems without both a correct and an incorrect pre-OPD response are retained in the full dataset denominator; zero here records unavailable preference information, not teacher equivalence, so these problems carry no directional evidence in this comparison.

Observable and unobservable regimes. The pairwise gap is directly estimable only when the pre-OPD sample contains at least one correct and one incorrect response. This condition holds for 103/130 problems in Qwen-Math, 158/542 on HumanEval+/MBPP+, 268/880 on LiveCodeBench v5, and 35/175 on LiveCodeBench v6 in Qwen-Code, 447/630 in DeepSeek-Math, and 213/800 in Qwen-Fact's four reported benchmarks. In particular, the score has no direct signal for zero-success problems, even though newly solving such problems can dominate a large-K outcome. Real OPD can change within-group response composition, share parameters across problems, use token-level approximations, and stop far from the restricted optimum.

Selection outcomes. TAS@K selects the higher-pass@K OPD-trained model in 148/177 setting-benchmark-budget comparisons: 40/44 for Qwen-Math, 28/36 for Qwen-Code, 48/53 for DeepSeek-Math, and 32/44 for Qwen-Fact, all comfortably above the 50% chance rate of a pairwise choice.

Top-k local proxy. We additionally evaluate a training-local proxy G that replaces the sampled-token log likelihood at each response position with the corresponding Base-weighted teacher log likelihood over the Base model's top-16 next-token candidates, then applies the same response-length normalization. Across the four matched pairs in Fig. 7, it obtains 39/44 Qwen-Math, 30/36 Qwen-Code, 48/53 DeepSeek-Math, and 29/44 Qwen-Fact point-direction agreements. Because it depends on a local truncation of the vocabulary and is not the realized finite update, we treat G as a sensitivity analysis and use the sampled-response token-average TAS@K in the main text.

B Evaluation Details

Finite-sample pass@K estimator. For each problem, we draw n responses and let c be the number judged correct. For any $K \le n$, we use the standard unbiased estimator (Chen et al., 2021)

$$\widehat{\text{pass@}K} = 1 - \frac{\binom{n-c}{K}}{\binom{n}{K}}, \tag{16}$$

with the ratio set to zero when $n - c < K$. Dataset-level pass@K averages this quantity over the problems in the evaluation set, yielding a single scalar per benchmark and sampling depth.

Math. We use the curated benchmark snapshots from `math-vault` (Ge, 2026b) and the `math-eval` pipeline (Ge, 2026a) for inference, replay, answer extraction, and scoring. Official metrics compare the last complete boxed answer $\boxed{\cdot}$ with the canonical answer. Unless otherwise noted, evaluation uses temperature 0.7, top-p 0.95, seed 0, a 32,768-token context window, a 1,024-token prompt limit, and a 31,744-token output limit. Each main Math curve uses 1,024 responses per problem on AMC2023, AIME2024, AIME2025, and AIME2026, as in the main text.

Math prompt format. For completion checkpoints, we pass the following string as a completion; for chat checkpoints, we place the identical string in the user message after the problem:

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

Code. HumanEval+ and MBPP+ are evaluated with the EvalPlus (Liu et al., 2023) task snapshots and unit-test harness[1], requiring a completion to pass the benchmark's base and additional tests. Both benchmarks use 512 responses per problem. LiveCodeBench v5 (880 problems) and v6 (175 new problems) use the official code-generation evaluator (Jain et al., 2025) with 128 responses per problem. All evaluations use the non-thinking Qwen3 chat serialization and seed 0. HumanEval+ and MBPP+ use temperature 0.7 and top-p 0.95; LiveCodeBench uses temperature 1.0 and top-p 0.8. Pass@K is computed from the per-problem correct-completion counts using the estimator above.

Code prompt formats. HumanEval+ and MBPP+ share the following user message, where `{{problem}}` is the benchmark-provided programming task. We supply no separate system message for these two benchmarks; the user message alone specifies the task and provides the problem.

HumanEval+ / MBPP+: user message

```
Solve the following programming problem.  Return only the complete
Python solution, without Markdown fences.  {{problem}}
```

LiveCodeBench v5 and v6 use the following system message. The user message depends on whether the task provides starter code; `{{question}}` denotes the problem statement.

LiveCodeBench: system message

```
You are an expert Python programmer.  You will be given a question
(problem specification) and will generate a correct Python program
that matches the specification and passes all tests.
```

LiveCodeBench: user message without starter code

```
### Question:
{{question}}

### Format:  Read the inputs from stdin solve the problem and write
the answer to stdout (do not directly test on the sample inputs).
Enclose your code within delimiters as follows.  Ensure that when the
python program runs, it reads the inputs, runs the algorithm and
writes output to STDOUT.
```python
# YOUR CODE HERE
```

Answer: (use the provided format with backticks)
```

[1] https://github.com/evalplus/evalplus

LiveCodeBench: user message with starter code

```
### Question:
{{question}}

### Format: You will use the following starter code to write the
solution to the problem and enclose your code within delimiters.
```python
{{starter_code}}
```

Answer: (use the provided format with backticks)
```

Fact. The closed-book Fact evaluation contains HotpotQA, TriviaQA, PopQA, and 2WikiMultiHopQA, with 200 problems per benchmark. The Fact training pool includes 90,447 HotpotQA train examples; for evaluation, we sampled 200 questions from the HotpotQA dev split after screening for exact question-text overlap with the full Fact training pool under NFKC normalization, case folding, and whitespace collapse; no matches were found. The SESA teacher is the released SESA-8B-search checkpoint (Fu et al., 2026b); its model page is https://huggingface.co/kuailexuexi/SESA-8B-search. This closed-book evaluation does not use the search tool from the original agentic setup. It uses the frozen non-thinking chat prompt, temperature 0.7, top-p 0.95, seed 0, a 4,096-token response cap, and an 8,192-token context cap. Responses are scored by exact match after answer extraction and normalization against the benchmark's accepted aliases; failed extractions count as incorrect. The shared pre-OPD base model, both OPD-trained models, and both teachers use 1,024 responses per problem on all four benchmarks, with a shared evaluation budget.

Fact prompt format. All four Fact benchmarks use the following user message without a separate system message. Only the question is substituted for {{problem}}; no retrieved context or gold answer is included.

Fact: user message

```
{{problem}}
Please reason step by step, then give a concise final answer in
<answer></answer> tags. Put the tagged answer on a new final line.
```

C Training Configurations

This appendix reports training configurations for the main experiments and OPD variants.

C.1 Main Experiments

Table 3 summarizes the configurations of our locally trained OPD models by domain. The Qwen3-1.7B-Base ← Qwen3-4B-Base-GRPO setting uses the released lllyx/Qwen3-1.7B-Base-OPD checkpoint (Li et al., 2026)[2].

Table 3: Training configurations by domain. Batch gives the global prompt batch size; length gives the prompt / response token limits; LR / schedule gives the learning rate and its schedule.

Domain	Training set	Batch	LR / schedule	Length
Math	DAPO-Math-17K (Yu et al., 2025)	64	10^{-6} / constant	1024 / 7168
Code	CodeR1-12K (ganler, 2025)	128	5×10^{-7} / cosine	2048 / 4096
Fact	NQ (Kwiatkowski et al., 2019) + HotpotQA	64	10^{-6} / constant	1024 / 3072

[2] https://huggingface.co/Thinking-Space/Qwen3-1.7B-Base-OPD

For reference, the model names in Table 1 are shorthands for the released checkpoints: Qwen3-4B-Base-GRPO denotes `Thinking-Space/Qwen3-4B-Base-GRPO`, Qwen3-4B-RL-Math denotes `Keven16/Qwen3-4B-Non-Thinking-RL-Math-Step500`, Qwen3-4B-RL-Code denotes `Keven16/Qwen3-4B-Non-Thinking-RL-Code-Step300`, Qwen3-8B-RL-RAR denotes `chentong00/Qwen3-8B-GRPO-Binary-RAR`, DS-Distill-Qwen-1.5B denotes `deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B`, JustRL-DS-1.5B denotes `hbx/JustRL-DeepSeek-1.5B` at global Step 260, and Skywork-OR1-Math-7B is used at Step 260 of `Skywork/Skywork-OR1-Math-7B`. The remaining names (Qwen3-1.7B-Base, Qwen3-1.7B, and Qwen3-8B) match their released checkpoints under the same names.

C.2 OPD Variants

OPD, Direct-OPD, EOPD, pure forward KL, and ExOPD are all trained using temperature 1, top-p 1, top-k disabled, one PPO epoch per batch, zero learning-rate warmup, and weight decay 0.01. Table 4 reports the configurations for this comparison, including per-variant batch sizes and lengths.

Table 4: Training configurations for OPD variants. Batch gives global / PPO mini-batch size; lengths give the maximum prompt and response lengths. Rollout sampling top-k is disabled for all locally executed rows. Each variant follows the default configuration of its respective repository.

Setting	Batch	Lengths	LR / schedule
OPD (Li et al., 2026)	64 / 64	1024 / 7168	1×10^{-6} / cosine
Direct-OPD (Feng et al., 2026)	128 / 32	1024 / 4096	3×10^{-6} / cosine
EOPD (Jin et al., 2026)	128 / 32	1024 / 4096	3×10^{-6} / cosine
FKL	128 / 32	1024 / 4096	3×10^{-6} / cosine
ExOPD (Yang et al., 2026b)	1024 / 1024	2048 / 16384	1×10^{-5} / constant

D Numerical Results for Test-Time Scaling Curves

Tables 5, 6, and 7 report benchmark-level pass@K (%) for the Math, Code, and Fact settings. Each setting identifies the student and teacher as student $\leftarrow$ teacher; Base, OPD, and Teacher occupy separate blocks, with one row per benchmark. Columns give the sampling budget K.

Table 5 lists the four Math settings on AMC2023 and AIME2024–2026. The benchmark-level values show where gains at pass@1 coexist with lower pass@1024 than the pre-OPD base model.

Table 6 gives the two Code settings on HumanEval+, MBPP+, and LiveCodeBench v5 and v6. It shows the Base–OPD comparison at each evaluated budget; HumanEval+ and MBPP+ reach $K = 512$, while both LiveCodeBench versions reach $K = 128$ in the reported evaluations.

Table 7 reports the two Fact settings on HotpotQA, TriviaQA, PopQA, and 2WikiMultiHopQA. In both settings, the OPD-trained model remains above its pre-OPD base model at pass@1024 on all four benchmarks; the large-budget comparison therefore favors the OPD-trained model.

Table 5: Math pass@K (%) for four student ← teacher settings on AMC2023 and AIME2024–2026. Rows compare Base, OPD, and Teacher; columns give the sampling budget K in each case.

Setting	Role	Dataset	1	2	4	8	16	32	64	128	256	512	1024
Qwen3-1.7B-Base ← Qwen3-4B-Base-GRPO	Base	AMC2023	32.3	44.4	56.3	66.8	75.8	83.5	89.7	94.3	97.7	99.6	100.0
		AIME2024	4.2	7.0	10.9	16.0	22.0	28.2	35.0	42.9	51.6	61.0	70.0
		AIME2025	3.1	5.3	8.3	12.4	17.7	23.5	29.6	37.0	45.7	55.1	66.7
		AIME2026	2.8	4.9	7.8	10.9	13.9	17.7	23.6	31.8	41.4	50.4	56.7
	OPD	AMC2023	45.4	55.4	63.9	71.7	78.7	83.4	86.6	89.2	91.0	92.5	95.0
		AIME2024	12.0	16.7	20.9	24.2	27.3	30.5	33.6	37.0	41.6	47.3	53.3
		AIME2025	8.6	12.6	17.4	22.2	26.5	30.4	34.4	39.0	44.6	51.2	56.7
		AIME2026	7.4	11.4	15.3	18.8	22.5	26.6	31.2	36.9	44.0	51.1	56.7
	Teacher	AMC2023	65.6	76.9	85.1	89.7	92.2	94.0	95.7	97.0	97.9	98.7	100.0
		AIME2024	24.2	30.7	37.2	44.4	51.8	58.2	63.1	67.1	70.4	73.7	76.7
		AIME2025	20.6	26.0	31.1	36.9	43.8	50.9	58.0	65.3	72.2	77.6	80.0
		AIME2026	18.7	24.6	30.1	34.8	39.8	45.2	51.0	57.7	65.1	72.0	76.7
DS-Distill-Qwen-1.5B ← Skywork-OR1-Math-7B	Base	AMC2023	72.2	82.5	89.8	93.6	95.3	96.2	97.2	98.5	99.6	100.0	100.0
		AIME2024	29.9	40.8	51.7	61.8	70.0	76.0	80.0	82.1	83.7	85.6	86.7
		AIME2025	23.4	29.3	34.6	40.2	46.2	52.1	58.0	64.1	69.5	73.6	76.7
		AIME2026	20.6	28.6	36.9	45.0	52.9	60.8	68.2	74.4	79.2	82.9	86.7
	OPD	AMC2023	76.0	85.2	91.6	94.3	95.1	95.4	95.8	96.4	97.1	97.5	97.5
		AIME2024	36.4	47.2	57.9	66.6	72.3	75.6	77.5	78.8	80.6	83.1	86.7
		AIME2025	28.5	33.4	37.7	42.0	46.4	50.8	55.1	60.0	64.7	67.9	70.0
		AIME2026	25.9	34.5	42.5	50.4	58.2	64.3	68.0	71.4	74.8	77.5	80.0
	Teacher	AMC2023	93.8	95.2	95.7	96.4	97.3	98.5	99.5	99.9	100.0	100.0	100.0
		AIME2024	67.0	75.4	80.2	82.6	84.1	85.5	86.7	87.9	88.9	89.8	90.0
		AIME2025	51.4	59.1	64.9	69.7	74.0	77.9	81.3	84.7	88.2	91.1	93.3
		AIME2026	60.9	68.9	74.9	79.2	82.1	84.5	86.6	88.5	89.7	90.0	90.0
DS-Distill-Qwen-1.5B ← JustRL-DS-1.5B	Base	AMC2023	72.2	82.5	89.8	93.6	95.3	96.2	97.2	98.5	99.6	100.0	100.0
		AIME2024	29.9	40.8	51.7	61.8	70.0	76.0	80.0	82.1	83.7	85.6	86.7
		AIME2025	23.4	29.3	34.6	40.2	46.2	52.1	58.0	64.1	69.5	73.6	76.7
		AIME2026	20.6	28.6	36.9	45.0	52.9	60.8	68.2	74.4	79.2	82.9	86.7
	OPD	AMC2023	87.6	92.9	94.9	95.6	96.2	97.1	98.4	99.5	100.0	100.0	100.0
		AIME2024	49.4	59.5	67.4	73.3	77.2	79.3	80.1	80.4	80.8	81.7	83.3
		AIME2025	34.7	40.6	46.4	52.2	57.3	61.2	64.5	67.5	69.8	71.6	73.3
		AIME2026	35.6	44.8	52.2	58.5	63.9	69.3	74.1	77.5	79.9	82.1	83.3
	Teacher	AMC2023	90.4	94.1	95.3	95.9	96.7	97.8	99.0	99.8	100.0	100.0	100.0
		AIME2024	52.1	61.9	70.1	76.1	79.3	80.2	80.4	80.8	81.7	83.3	86.7
		AIME2025	36.5	42.8	49.4	55.8	60.7	63.8	66.2	68.4	70.8	73.8	76.7
		AIME2026	38.0	47.1	54.2	59.6	64.4	69.7	74.2	77.1	79.0	79.9	80.0
Qwen3-1.7B ← Qwen3-4B-RL-Math	Base	AMC2023	46.1	59.2	70.6	79.2	85.2	89.6	93.2	96.1	98.1	99.5	100.0
		AIME2024	13.7	19.3	25.0	31.0	37.8	45.0	51.3	56.4	61.7	68.2	76.7
		AIME2025	10.1	14.1	18.0	22.3	27.7	34.3	41.6	48.5	55.7	63.2	70.0
		AIME2026	9.1	13.3	17.3	21.1	25.8	31.1	36.5	42.3	49.9	59.8	70.0
	OPD	AMC2023	82.5	88.9	92.6	94.6	95.5	96.0	96.6	97.2	97.5	97.5	97.5
		AIME2024	40.2	48.7	56.1	62.2	66.5	69.9	74.1	78.6	82.1	84.6	86.7
		AIME2025	33.1	39.8	46.7	52.5	57.3	62.2	67.2	70.8	73.0	74.8	76.7
		AIME2026	33.5	41.6	48.3	53.8	57.7	60.5	62.8	65.1	67.2	69.8	73.3
	Teacher	AMC2023	94.7	96.3	97.0	97.4	97.5	97.5	97.5	97.5	97.5	97.5	97.5
		AIME2024	61.2	70.8	77.1	80.2	82.0	83.3	84.4	85.4	86.7	88.2	90.0
		AIME2025	57.8	67.2	73.0	76.4	78.5	80.0	81.2	82.7	84.8	87.9	90.0
		AIME2026	57.8	64.6	70.2	74.3	76.8	78.8	80.7	82.5	84.7	87.5	90.0

Table 6: Code pass@K (%) for two student ← teacher settings on HumanEval+, MBPP+, and LiveCodeBench v5 and v6. Rows compare Base, OPD, and Teacher; columns give K, with dashes marking entries that were not evaluated for the corresponding checkpoint and benchmark.

Setting	Role	Dataset	1	2	4	8	16	32	64	128	256	512	1024
Qwen3-1.7B ← Qwen3-8B	Base	HumanEval+	56.2	60.3	63.6	66.6	69.2	71.1	72.3	72.9	73.4	73.8	—
		MBPP+	48.6	51.4	53.5	55.3	56.9	58.1	59.1	59.9	60.6	61.4	—
		LiveCodeBench v5	26.0	29.8	33.3	36.2	38.8	41.1	43.1	44.8	—	—	—
		LiveCodeBench v6	15.5	17.7	19.6	21.4	23.2	25.1	27.1	29.1	—	—	—
	OPD	HumanEval+	58.9	63.9	68.3	72.1	75.2	77.6	79.2	80.2	81.0	81.7	—
		MBPP+	49.6	53.5	56.6	58.9	61.1	62.9	64.4	65.7	66.9	68.0	—
		LiveCodeBench v5	27.9	32.7	37.1	41.1	44.8	48.2	51.4	54.1	—	—	—
		LiveCodeBench v6	18.9	22.0	24.7	27.2	29.2	30.9	32.6	34.3	—	—	—
	Teacher	HumanEval+	78.3	80.0	81.3	82.4	83.6	84.8	85.8	86.6	87.4	87.8	—
		MBPP+	68.0	69.7	71.2	72.6	73.8	74.7	75.3	75.6	75.8	75.9	—
		LiveCodeBench v5	43.2	48.4	52.9	56.6	59.8	62.6	64.9	67.0	—	—	—
		LiveCodeBench v6	25.8	28.6	31.2	33.2	34.8	36.3	38.0	39.4	—	—	—
Qwen3-1.7B ← Qwen3-4B-RL-Code	Base	HumanEval+	56.2	60.3	63.6	66.6	69.2	71.1	72.3	72.9	73.4	73.8	—
		MBPP+	48.6	51.4	53.5	55.3	56.9	58.1	59.1	59.9	60.6	61.4	—
		LiveCodeBench v5	26.0	29.8	33.3	36.2	38.8	41.1	43.1	44.8	—	—	—
		LiveCodeBench v6	15.5	17.7	19.6	21.4	23.2	25.1	27.1	29.1	—	—	—
	OPD	HumanEval+	62.0	70.5	77.3	82.5	86.1	88.4	89.9	90.9	91.6	92.1	—
		MBPP+	50.8	56.5	61.2	65.1	68.4	71.3	74.0	76.5	78.6	80.2	—
		LiveCodeBench v5	36.1	43.9	50.4	55.6	59.6	63.0	65.8	68.1	—	—	—
		LiveCodeBench v6	21.7	26.0	30.0	33.2	35.5	37.2	38.6	40.0	—	—	—
	Teacher	HumanEval+	79.6	82.7	85.1	87.2	88.9	90.1	90.9	91.6	92.0	92.1	—
		MBPP+	65.5	68.5	70.9	72.7	74.2	75.3	76.0	76.5	77.0	77.5	—
		LiveCodeBench v5	53.3	61.5	67.6	71.8	74.7	76.8	78.6	80.1	—	—	—
		LiveCodeBench v6	32.1	36.5	39.7	42.1	44.5	46.7	48.8	50.9	—	—	—

Table 7: Fact pass@K (%) for two student ← teacher settings on HotpotQA, TriviaQA, PopQA, and 2WikiMultiHopQA. Rows compare Base, OPD, and Teacher; columns give the sampling budget K.

Setting	Role	Dataset	1	2	4	8	16	32	64	128	256	512	1024
Qwen3-1.7B ← Qwen3-8B-RL-RAR	Base	HotpotQA	5.6	7.5	9.6	11.8	13.9	15.8	17.8	19.7	21.5	23.4	25.0
		TriviaQA	21.1	24.8	28.2	31.4	34.7	38.1	41.4	44.5	47.1	49.2	51.0
		PopQA	8.3	9.9	11.5	13.3	15.0	16.8	18.6	20.3	22.1	23.6	24.5
		2WikiMultiHopQA	3.0	4.4	5.9	7.3	8.7	10.0	11.2	12.3	13.3	14.5	16.0
	OPD	HotpotQA	5.2	7.7	10.5	13.4	16.3	19.3	22.2	24.8	27.1	29.3	32.0
		TriviaQA	22.4	27.4	32.0	36.2	40.4	44.2	47.4	50.0	52.5	55.4	58.5
		PopQA	10.1	12.1	14.2	16.6	19.3	22.1	25.0	27.9	30.6	32.8	34.5
		2WikiMultiHopQA	4.4	6.1	7.9	9.9	11.9	14.0	16.3	18.9	21.6	24.5	27.0
	Teacher	HotpotQA	10.5	13.1	16.3	19.9	23.4	26.7	29.9	33.0	36.1	38.9	41.0
		TriviaQA	41.9	47.6	52.3	56.4	59.9	63.2	66.2	68.9	71.2	73.2	75.0
		PopQA	14.1	17.3	20.5	23.3	25.9	28.4	30.8	33.1	35.4	37.7	40.5
		2WikiMultiHopQA	5.0	7.0	9.3	11.9	14.3	16.7	19.3	22.0	24.7	27.7	30.5
Qwen3-1.7B ← SESA-8B-search	Base	HotpotQA	5.6	7.5	9.6	11.8	13.9	15.8	17.8	19.7	21.5	23.4	25.0
		TriviaQA	21.1	24.8	28.2	31.4	34.7	38.1	41.4	44.5	47.1	49.2	51.0
		PopQA	8.3	9.9	11.5	13.3	15.0	16.8	18.6	20.3	22.1	23.6	24.5
		2WikiMultiHopQA	3.0	4.4	5.9	7.3	8.7	10.0	11.2	12.3	13.3	14.5	16.0
	OPD	HotpotQA	5.6	8.0	10.7	13.5	16.2	18.8	21.6	24.6	27.5	30.4	33.0
		TriviaQA	22.4	27.0	31.0	34.8	38.6	42.3	45.5	48.6	51.5	54.3	56.5
		PopQA	11.7	14.0	16.5	19.1	21.8	24.5	27.1	29.7	32.2	34.4	36.5
		2WikiMultiHopQA	3.3	5.0	6.9	8.9	10.9	12.8	14.8	17.1	19.7	22.3	24.5
	Teacher	HotpotQA	12.4	15.5	18.8	22.2	25.6	28.8	32.0	35.4	39.2	42.9	46.0
		TriviaQA	43.4	48.4	52.7	56.2	58.9	61.2	63.6	65.9	67.8	69.7	72.0
		PopQA	16.5	19.7	22.7	25.6	28.5	31.4	34.5	37.4	40.2	43.0	46.5
		2WikiMultiHopQA	5.7	8.0	10.6	13.2	15.6	18.0	20.6	23.3	25.9	28.1	30.0

E ADDITIONAL AVG@K CURVES

Figs. 12, 13, and 14 complement the pass@K curves in Figs. 4, 2, and 3, using the same evaluation runs. For each problem, avg@K is the fraction of correct responses among its first K samples; we then average over problems. It measures sampled-response correctness, whereas pass@K measures whether at least one of K responses is correct. At each panel's maximum budget $K = n$, avg@n equals the finite-sample pass@1 estimate computed from all n responses. Avg@1 uses only the first response per problem and therefore need not equal the pass@1 estimate reported above.

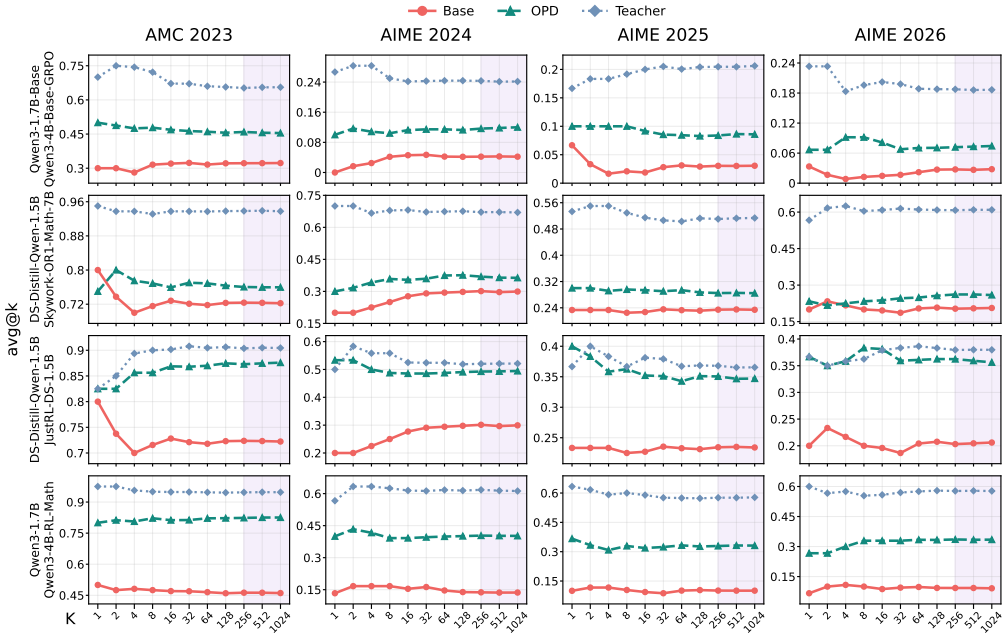


Figure 12: **Avg@K in four Math settings.** Each point averages the correct-response fraction in the first K samples per problem. Rows identify student–teacher settings; columns show AMC2023 and AIME2024–2026. Curves compare Base, OPD, and Teacher up to $K = 1024$.

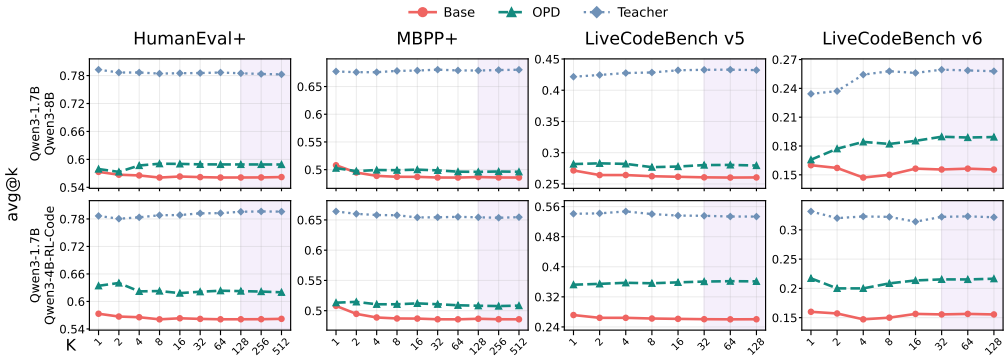


Figure 13: **Avg@K in two Code settings.** Rows identify student–teacher settings, and columns show HumanEval+, MBPP+, and LiveCodeBench v5 and v6. Curves compare Base, OPD, and Teacher; the maximum K is 512 for HumanEval+ and MBPP+ and 128 for LiveCodeBench.

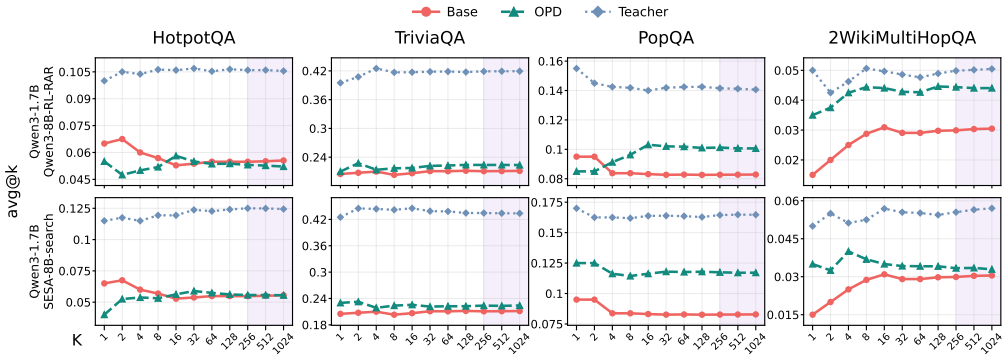


Figure 14: **Avg@K in two Fact settings.** Rows use Qwen3-8B-RL-RAR and SESA-8B-search as teachers for Qwen3-1.7B; columns show HotpotQA, TriviaQA, PopQA, and 2WikiMultiHopQA. Curves compare Base, OPD, and Teacher up to $K = 1024$. On HotpotQA with the RAR teacher, OPD has slightly lower avg@1024 than Base, although its pass@1024 is higher, so the two metrics order the two models differently at this sampling budget (Table 7).

F Random-Trajectory Perplexity Analysis

To construct Fig. 10, we use evaluation runs for the pre-OPD base, OPD-trained, and teacher models in the DS-Distill-Qwen-1.5B ← Skywork-OR1-Math-7B setting on AMC2023, AIME2024, AIME2025, and AIME2026. With seed 0, we independently sample four problems from each benchmark, yielding 16 problems in total. For every selected problem, we sample 32 trajectories from each source model, resulting in $16 \times 32 = 512$ trajectories per source and 1,536 trajectories overall.

Under the teacher scorer, trajectories generated by the OPD-trained model have lower perplexity than those generated by the pre-OPD base model. Under the pre-OPD base scorer, trajectories generated by the OPD-trained model have lower perplexity than those generated by either the pre-OPD base model or the teacher. Together, these observations suggest that OPD shifts the student's trajectory distribution toward paths favored by the teacher while remaining supported by the pre-OPD base model, rather than simply reproducing the teacher's reasoning trajectories verbatim.

G Numerical Results for Training Dynamics

Tables 8 and 9 give benchmark-level pass@K (%) for the checkpoints shown in Fig. 8 and 9, respectively. Checkpoint blocks list Base and each evaluated OPD training step. Math uses 1,024 responses per problem; Code sampling depth varies by benchmark and checkpoint, so dashes mark budgets that were not evaluated for the corresponding checkpoint and benchmark.

Table 8 expands the DS-Distill-Qwen-1.5B ← Skywork-OR1-Math-7B training trajectory into pass@K values for each benchmark and evaluated training step. At Step 260, pass@1 exceeds that of the pre-OPD base model on all four benchmarks, whereas pass@1024 becomes lower on AMC2023, AIME2025, and AIME2026, and matches the pre-OPD base model on AIME2024. By Step 80, pass@1024 has already dropped below the pre-OPD base model on three benchmarks, including a 10.0 pp decrease on AIME2026; however, the subsequent checkpoints do not exhibit a monotonic decline. These results show that a pass@1 gain can coexist with lower pass@1024, and that the decrease can appear early in OPD training, well before the final training checkpoint.

Table 9 expands the Qwen3-1.7B ← Qwen3-4B-RL-Code training trajectory into pass@K values for HumanEval+, MBPP+, and LiveCodeBench v5 and v6. Already at Step 100, pass@1 and the largest evaluated budget for each benchmark exceed the corresponding Base values. The available sampling depth varies across benchmark and checkpoint rows, so not every budget is available.

Table 8: Math training dynamics: pass@K (%) for Base and Steps 20–260 on AMC2023 and AIME2024–2026, corresponding to Fig. 8. Columns give the sampling budget K.

Setting	Checkpoint	Dataset	1	2	4	8	16	32	64	128	256	512	1024
DS-Distill-Qwen-1.5B ← Skywork-OR1-Math-7B	Base	AMC2023	72.2	82.5	89.8	93.6	95.3	96.2	97.2	98.5	99.6	100.0	100.0
		AIME2024	29.9	40.8	51.7	61.8	70.0	76.0	80.0	82.1	83.7	85.6	86.7
		AIME2025	23.4	29.3	34.6	40.2	46.2	52.1	58.0	64.1	69.5	73.6	76.7
		AIME2026	20.6	28.6	36.9	45.0	52.9	60.8	68.2	74.4	79.2	82.9	86.7
	Step 20	AMC2023	73.3	83.3	90.3	94.0	95.5	96.3	97.3	98.6	99.7	100.0	100.0
		AIME2024	31.8	43.0	54.1	63.9	71.5	76.7	79.7	81.3	82.3	83.1	83.3
		AIME2025	24.7	30.5	36.0	41.7	47.8	53.4	58.6	63.7	69.1	73.6	76.7
		AIME2026	22.6	31.1	39.7	47.9	55.9	63.6	70.3	75.6	79.9	83.3	86.7
	Step 80	AMC2023	68.3	77.4	85.3	90.6	92.9	94.0	94.8	95.5	96.1	96.9	97.5
		AIME2024	32.1	42.5	52.8	62.0	69.5	74.9	78.5	81.2	83.9	86.1	86.7
		AIME2025	26.2	31.8	36.4	40.7	45.3	49.9	54.5	59.1	63.9	67.7	70.0
		AIME2026	23.7	31.5	38.8	45.6	53.2	60.7	66.3	70.7	74.4	76.2	76.7
	Step 140	AMC2023	72.9	82.4	89.9	93.7	94.9	95.1	95.2	95.3	95.6	96.2	97.5
		AIME2024	34.9	45.5	56.1	65.1	71.5	75.3	77.4	78.8	80.5	82.5	83.3
		AIME2025	27.6	32.8	37.2	41.5	46.0	50.4	54.6	58.9	63.0	66.6	70.0
		AIME2026	24.7	33.1	41.4	49.5	57.7	64.4	68.6	71.8	74.5	76.7	80.0
	Step 200	AMC2023	74.1	83.4	90.5	94.0	95.0	95.1	95.2	95.3	95.6	96.2	97.5
		AIME2024	35.3	46.3	57.2	66.1	71.6	74.7	76.7	78.8	81.2	84.0	86.7
		AIME2025	27.7	32.6	36.8	41.0	45.4	50.1	55.1	60.1	64.8	68.5	70.0
		AIME2026	24.0	32.3	40.5	48.7	57.0	63.7	68.0	71.4	75.0	79.6	86.7
	Step 260	AMC2023	76.0	85.2	91.6	94.3	95.1	95.4	95.8	96.4	97.1	97.5	97.5
		AIME2024	36.4	47.2	57.9	66.6	72.3	75.6	77.5	78.8	80.6	83.1	86.7
		AIME2025	28.5	33.4	37.7	42.0	46.4	50.8	55.1	60.0	64.7	67.9	70.0
		AIME2026	25.9	34.5	42.5	50.4	58.2	64.3	68.0	71.4	74.8	77.5	80.0

Table 9: Code training dynamics: pass@K (%) for Base and Steps 100–500 on HumanEval+, MBPP+, and LiveCodeBench v5 and v6, corresponding to Fig. 9. Columns give K; dashes mark budgets that were not evaluated for the corresponding checkpoint and benchmark.

Setting	Checkpoint	Dataset	1	2	4	8	16	32	64	128	256	512	1024
Qwen3-1.7B ← Qwen3-4B-RL-Code	Base	HumanEval+	56.2	60.3	63.6	66.6	69.2	71.1	72.3	72.9	73.4	73.8	—
		MBPP+	48.6	51.4	53.5	55.3	56.9	58.1	59.1	59.9	60.6	61.4	—
		LiveCodeBench v5	26.0	29.8	33.3	36.2	38.8	41.1	43.1	44.8	—	—	—
		LiveCodeBench v6	15.5	17.7	19.6	21.4	23.2	25.1	27.1	29.1	—	—	—
	Step 100	HumanEval+	59.5	67.0	73.4	78.9	83.2	85.9	87.5	88.5	89.6	—	—
		MBPP+	50.8	56.0	60.5	64.4	67.8	70.8	73.6	76.1	78.0	—	—
		LiveCodeBench v5	34.7	41.9	48.2	53.4	57.7	61.1	63.9	66.0	—	—	—
		LiveCodeBench v6	21.5	25.2	28.6	31.6	34.1	36.4	38.5	40.6	—	—	—
	Step 200	HumanEval+	61.1	68.9	75.7	81.3	85.1	87.3	88.7	89.8	90.9	—	—
		MBPP+	50.4	55.7	60.3	64.3	67.8	70.8	73.4	75.4	76.7	—	—
		LiveCodeBench v5	35.4	43.0	49.4	54.7	58.9	62.4	65.5	68.0	—	—	—
		LiveCodeBench v6	21.2	25.4	29.3	32.4	34.7	36.5	38.2	39.4	—	—	—
	Step 300	HumanEval+	61.7	69.9	76.7	82.0	85.6	88.0	89.8	91.0	91.5	—	—
		MBPP+	50.3	56.0	60.8	64.8	68.0	70.8	73.5	75.9	77.8	—	—
		LiveCodeBench v5	35.8	43.6	50.3	55.7	60.0	63.6	66.7	69.2	—	—	—
		LiveCodeBench v6	21.1	25.6	29.7	32.9	35.1	36.7	38.0	38.9	—	—	—
	Step 400	HumanEval+	62.0	70.5	77.3	82.5	86.1	88.4	89.9	90.9	91.6	92.1	—
		MBPP+	50.8	56.5	61.2	65.1	68.4	71.3	74.0	76.5	78.6	80.2	—
		LiveCodeBench v5	36.1	43.9	50.4	55.6	59.6	63.0	65.8	68.1	—	—	—
		LiveCodeBench v6	21.7	26.0	30.0	33.2	35.5	37.2	38.6	40.0	—	—	—
	Step 500	HumanEval+	62.6	71.5	78.3	83.3	86.8	89.3	90.7	91.8	92.7	—	—
		MBPP+	50.4	56.1	60.8	64.7	67.8	70.4	73.1	75.9	78.6	—	—
		LiveCodeBench v5	37.0	44.8	51.2	56.4	60.4	63.7	66.3	68.4	—	—	—
		LiveCodeBench v6	22.2	26.6	30.5	33.6	35.9	37.6	39.0	40.0	—	—	—