Rethinking the Capacity Gap in On-Policy Distillation for Large Language Models

Anonymous ACL submission

Abstract

On-policy distillation (OPD) is increasingly adopted in modern post-training pipelines as a remedy for the exposure bias and catastrophic forgetting of supervised fine-tuning. Yet stronger teachers still frequently fail to improve, and sometimes actively degrade, the student under OPD. We dissect OPD's training signal from three different perspectives to explain *what* it learns, *where* its gains come from, and *why* it sometimes hurts. *(i) What:* OPD transfers the teacher's *uncertainty profile* rather than its problem-level knowledge. *(ii) Where:* gains come from aligning the student's per-position *entropy shape* with the teacher's, so already shape-aligned pairings have no headroom to gain. *(iii) Why:* the regression under OPD is caused by the teacher pulling the student off confidently-correct tokens, which triggers catastrophic forgetting during training. Together, these findings give a unified mechanistic account of when and why OPD helps or hurts, and some probe methods that predicts the likely outcome of OPD before training begins.

1 Introduction

Distillation has emerged as an effective technique for compressing strong models and transferring their knowledge into compact students (Hinton et al., 2015). However, prior work has shown that stronger models are not always optimal teachers for off-policy distillation approaches such as supervised fine-tuning (SFT) (Xu et al., 2025; Li et al., 2025; Zhang et al., 2025b). In these methods, the student is trained on fixed sequences generated by the teacher, creating a distribution mismatch (i.e., exposure bias) between the trajectories seen during training and those produced autoregressively at inference (Ross and Bagnell, 2010; Bengio et al., 2015). Such off-policy supervision can also induce catastrophic forgetting, causing the student's performance to fall below its pre-distillation baseline (Yang et al., 2024; Kajitsuka et al., 2026).

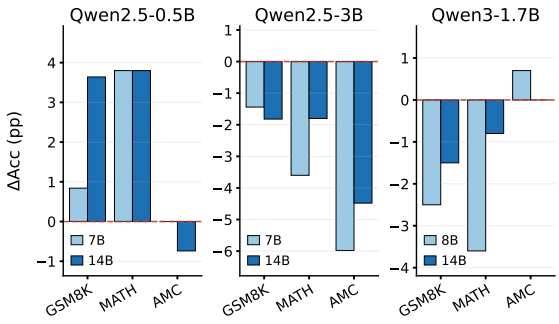

Figure 1: **Stronger teachers can fail to improve, and may even degrade, the student under on-policy distillation.** We report ΔAccuracy (in absolute percentage points, pp) of OPD-trained students relative to their pre-distillation baseline on three math reasoning benchmarks. All models are INSTRUCT versions, and the teachers are from the same model family as the student.

On-policy distillation (OPD) offers a promising alternative and has been adopted in industrial post-training pipelines, including Qwen3 (Yang et al., 2025), MiMo (Xiao et al., 2026), GLM-5 (Zeng et al., 2026). Unlike off-policy distillation, OPD obtains token-level supervision from the teacher on trajectories generated by the student itself, thereby training the student on states it actually visits at inference time (Gu et al., 2024; Agarwal et al., 2024). Recent studies further suggest that on-policy data can mitigate catastrophic forgetting, highlighting the potential of OPD for continual learning (Chen et al., 2025; Lu and Lab, 2025).

Despite these advances, OPD still suffers from the same issue, as shown in Figure 1: a stronger teacher can fail to improve a student or even degrade its performance. Concurrent work investigates when and why OPD succeeds or fails, identifying two empirical patterns associated with its effectiveness (Li et al., 2026). Yet their token-level mechanistic analysis does not fully explain the underlying mechanisms of OPD, leaving open three deeper questions: *what* does OPD actually learn,

where do its gains come from, and *why* does it sometimes degrade the student?

To answer these questions, we present a systematic analysis from three different perspectives: *data*, per-token *entropy*, and per-token *reward*.

Data (correctness): *what* does OPD learn? We control for teacher knowledge by training on prompts where two candidate teachers *both succeed* or *both fail* on the final answer. In either case OPD yields comparable downstream accuracy, indistinguishable from training on randomly sampled prompts (§5.1). Knowledge transfer is therefore *not* the primary mechanism of OPD: what is transferred is the teacher's *reasoning behavior*, not its problem-level correctness.

Token (entropy): *where* do its gains come from? We quantify the alignment of student and teacher *uncertainty profiles* along the rollout with two shape metrics—per-position entropy correlation ρ and top-20% fork-token overlap—and one magnitude metric, the entropy-value gap ΔH. The pre-OPD value of the two shape metrics cleanly predicts whether OPD will help or hurt, while ΔH does not (§5.2). OPD's gains come from closing the *shape* gap, which leaves no headroom when the student and teacher are already shape-aligned—precisely the regime where a stronger same-family teacher fails or degrades the student.

Token (reward): *why* does it sometimes degrade? We decompose the per-token OPD reward $r_t = \log q_t(\hat{y}_t) - \log p_t(\hat{y}_t)$ jointly by the sign of r_t and the student's entropy at the same position, isolating the *destructive* Q3 population: low-entropy positions on *correct* rollouts at which the teacher pulls the student off tokens it already had right. The destructive-to-corrective force ratio DEST:CORR cleanly separates the regressing pairings (>1) from the improving ones (<1), and a targeted ablation that masks these low-entropy positions causally suppresses the regression (§5.3). Destructive Q3 tokens is thus the main cause of catastrophic forgetting under OPD.

Contributions.

- We empirically demonstrate that the failure of stronger teachers persists under on-policy distillation across the Qwen2.5 and Qwen3 model families, including cases where distillation drives the student below its pre-distillation performance (§4).
- Through a systematic mechanistic analysis (§5) we show that *(i)* OPD transfers the teacher's *reasoning behavior* rather than its problem-level knowledge; *(ii)* OPD's gains come from aligning the student's per-position *entropy shape* with the teacher's, so pairings that are already shape-aligned have no headroom; and *(iii)* the regression is caused by *destructive Q3 tokens*, teacher disagreements at confidently-correct tokens.

2 Related Work

On-Policy Distillation. Exposure bias in off-policy distillation motivates the shift to on-policy distillation. MiniLLM (Gu et al., 2024) introduces OPD for LLMs, optimizing the reverse KL objective on student-generated rollouts via policy gradients, while GKD (Agarwal et al., 2024) generalizes this approach by directly minimizing token-level divergences on mixtures of on- and off-policy data. Thinking Machines Lab (Lu and Lab, 2025) demonstrate that on-policy dense supervision is more efficient than outcome-reward RL and shows promise for continual learning. More recently, OPD has been extended to the self-distillation setting, where the student serves as its own teacher by conditioning on privileged information (Hübotter et al., 2026; Shenfeld et al., 2026; Zhao et al., 2026). Concurrent studies have begun to investigate the failure modes and underlying mechanisms of OPD (Li et al., 2026; Fu et al., 2026). Our work further contributes to this direction through an in-depth analysis of the root causes underlying OPD success and failure.

Curse of Capacity Gap. A widely observed phenomenon in knowledge distillation is that large teacher–student capacity gaps can diminish or even reverse the benefits of distillation (Cho and Hariharan, 2019; Mirzadeh et al., 2020). Zhang et al. (2023) document the *curse of capacity gap* in language model distillation, and subsequent work extends this observation to instruction tuning (Xu et al., 2025) and chain-of-thought distillation (Li et al., 2025). A body of work seeks to address this issue through better teacher selection. Zhang et al. (2025a) characterize the relationship between student scale and optimal teacher scale, while Busbridge et al. (2025) derive general distillation scaling laws for predicting student performance. Another line of research aims to bridge the teacher-student gap by constructing more suit-

able teacher. For example, Ding et al. (2025) employ intermediate-sized model as teacher assistants, and Zhang et al. (2025b) achieve personalized distillation through a query-level teacher router. However, existing studies mainly focus on off-policy knowledge distillation, leaving the capacity-gap issue in OPD underexplored.

3 Preliminaries

3.1 Notation

Let $x \in \mathcal{D}_x$ denote a query and $y = (y_1, \ldots, y_T)$ be a response of length T . We write $y_{<t} = (y_1, \ldots, y_{t-1})$ for the prefix up to step t. A language model π defines a next-token distribution $\pi(\cdot \mid x, y_{<t})$ over a vocabulary $\mathcal{V}$. We refer to the model being trained as the *student* π_θ, with learnable parameters θ, and to a frozen reference model as the *teacher* π_T. Both models share the same vocabulary $\mathcal{V}$. Knowledge distillation transfers knowledge from π_T to π_θ by minimizing the KL divergence between the distributions $\mathcal{P}$ and $\mathcal{Q}$:

$$D_{\mathrm{KL}}(\mathcal{P} \,\|\, \mathcal{Q}) = \sum_{v \in \mathcal{V}} \mathcal{P}(v) \log \frac{\mathcal{P}(v)}{\mathcal{Q}(v)}. \quad (1)$$

3.2 On-Policy Distillation

Given a query $x \sim \mathcal{D}_x$, *On-policy Distillation* first samples a response $y \sim \pi_\theta(\cdot \mid x)$, and then the student-generated trajectory is forwarded through both π_θ and π_T, yielding two next-token distributions for each prefix $y_{<t}$: $p_t = \pi_\theta(\cdot \mid x, y_{<t})$ and $q_t = \pi_T(\cdot \mid x, y_{<t})$. The standard OPD objective is the sequence-level reverse KL divergence between the student and teacher under trajectories sampled from the student policy, which, by autoregressive factorization, can be formalized as a token-level KL objective:

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{x \sim \mathcal{D}_x,\, y \sim \pi_\theta(\cdot|x)} \left[\sum_{t=1}^{T} D_{\mathrm{KL}}(p_t \,\|\, q_t) \right]. \quad (2)$$

Following Agarwal et al. (2024), we do not backpropagate through the student's sampling distribution $\pi_\theta(\cdot \mid x)$ that produces y. Gradients are computed only through π_θ in the per-token KL terms, thereby avoiding policy-gradient optimization (Williams, 1992), which leads to more stable training.

In practice, there are three common implementations that differ in how the per-token reverse KL is computed. *Sampled-token OPD* is the most lightweight variant, evaluating only the token sampled by the student at each step and yielding an unbiased Monte Carlo estimator of each token-level KL term. *Full-vocabulary OPD*, at the other extreme, computes the KL divergence over the entire vocabulary at each prefix. *Top-k OPD* provides an intermediate design by restricting the divergence computation to a subset of full vocabulary.

Here we focus on the student top-k variant, as it achieves a good trade-off between supervision fidelity and computational cost. Specifically, at each prefix $y_{<t}$, we select the subset $S_t = \mathrm{TopK}(p_t, k)$, consisting of the k tokens with the highest probabilities under the student distribution p_t. We then restrict both the student and teacher distributions to S_t and renormalize them as follows:

$$\bar{p}_t^{(S_t)} = \frac{p_t \odot \mathbf{1}_{S_t}}{\sum_{u \in S_t} p_t(u)}, \qquad \bar{q}_t^{(S_t)} = \frac{q_t \odot \mathbf{1}_{S_t}}{\sum_{u \in S_t} q_t(u)}, \quad (3)$$

Distillation is then performed by minimizing the subset KL divergence $D_{\mathrm{KL}}\big(\bar{p}_t^{(S_t)} \,\|\, \bar{q}_t^{(S_t)}\big)$. This yields an approximation to the full-vocabulary reverse KL, where probability mass outside S_t is discarded.

4 Capacity Gap in On-Policy Distillation

A recurring observation in traditional off-policy distillation is that stronger teachers do not always yield better students, a phenomenon known as the capability gap. Since on-policy distillation has been shown to effectively mitigate exposure bias and catastrophic forgetting, a natural question is whether it can also overcome the capability gap.

Datasets and Evaluation. We conduct our OPD experiments on 10,000 prompts randomly sampled from the OpenR1-Math-220k dataset[1]. Unless otherwise specified, we use the default implementation and hyperparameter settings described in Appendix A. The distilled student models are evaluated on **GSM8K** (Cobbe et al., 2021), **MATH-500** (Lightman et al., 2024), and **AMC** (Li et al., 2024). Results on **AIME24/25** (Balunovic et al., 2026) are provided as complementary evaluations for strong student models. We use the *EvalScope* framework for standardized evaluation across all benchmarks and report Pass@1 accuracy averaged over three independent runs.

Models. We study three student models spanning two model families: **Qwen2.5-0.5B-Instruct**, **Qwen2.5-3B-Instruct** (Qwen et al., 2025), and **Qwen3-1.7B (Non-thinking)** (Yang et al., 2025).

[1] https://huggingface.co/datasets/open-r1/OpenR1-Math-220k

Setup	GSM8K	MATH-500	AMC
Qwen2.5-0.5B *(baseline)*	44.9	29.0	9.0
+ 3B	**46.9**	**30.0**	8.9
+ 7B	**45.7**	**32.8**	9.0
+ 14B	**48.5**	**32.8**	8.2
Qwen2.5-3B *(baseline)*	85.8	63.8	33.6
+ 7B	84.4	60.2	27.6
+ 14B	84.0	62.0	29.1
Qwen3-1.7B *(baseline)*	82.9	73.6	40.3
+ Qwen3-4B	79.8	71.0	35.8
+ Qwen3-8B	80.4	70.0	**41.0**
+ Qwen3-14B	81.4	72.8	40.3

Table 1: Accuracy of OPD-trained students across different student-teacher setups. Each block begins with the pre-distillation *baseline*, followed by OPD results with same-family teachers of different sizes. We **bold** numbers that show improvements from distillation. Results on AIME24/25 are reported in Appendix B.

For each student, we sweep teachers of increasing size from the same family: 3B/7B/14B for Qwen2.5 students, and 4B/8B/14B for the Qwen3 student[2].

Results. Table 1 presents that stronger teachers may fail to improve the student and can even degrade its performance. For both *Qwen2.5-3B-Instruct* and *Qwen3-1.7B*, performance degradation is consistently observed across all teacher choices. These results empirically demonstrate that the capability gap persists in OPD, consistent with observations from concurrent work (Li et al., 2026).

Unlike the conventional off-policy setting, where larger teacher-student capacity gaps usually lead to worse distillation outcomes, OPD shows a different pattern. For example, *Qwen2.5-0.5B-Instruct* achieves modest improvements on GSM8K and MATH-500 even with larger teachers. Moreover, for *Qwen2.5-3B-Instruct* and *Qwen3-1.7B-Instruct*, the degree of performance degradation tends to decrease as teacher size increases. These findings motivate us to identify and understand the distinctive mechanisms underlying the capability gap in OPD.

5 Mechanistic Analysis

Having established in §4 that stronger teachers frequently fail or even degrade the student under OPD, we now ask *why*. We dissect OPD's training signal through three complementary analyses, each answering one diagnostic question about a different aspect of the signal: *(i)* a correctness-controlled ablation (§5.1) asks *what* OPD actually learns; *(ii)* a per-token entropy analysis (§5.2) asks *where* its gains come from; and *(iii)* a per-token reward analysis (§5.3) asks *why* it sometimes degrades the student. Figure 2 summarizes the question and main finding of each subsection.

5.1 Data: Supervision Quality Analysis

In off-policy distillation, such as SFT, an implicit assumption is that the pre-collected data have been filtered for correctness, so the student learns primarily from correct reasoning paths. However, OPD provides no such guarantee. On the same dataset, stronger teachers may provide more correct supervision, whereas smaller teachers may themselves fail on many problems. This raises a natural question: *what does OPD actually learn?* If OPD mainly learns from correctness, this could explain the benefits sometimes observed with larger teacher-student capacity gaps, given the substantial knowledge gap between stronger and smaller teachers.

Setup. We investigate this question by designing rigorous controlled experiments. Using the same math reasoning dataset as in §4, we construct two correctness-controlled subsets of 7,000 prompts each, where a prompt is deemed *correct* for a teacher if at least one of four rollouts at temperature 1.0 yields the correct final answer. The two subsets are: (1) **both-correct**, on which both *Qwen2.5-7B-Instruct* and *Qwen2.5-14B-Instruct* succeed; and (2) **both-wrong**, on which both teachers fail.

By construction, the effect of knowledge gap between different teachers are ablated. We then conduct OPD on each (student, teacher) pair across all three subsets: the original 10k random sample and the two controlled subsets.

Results. The results are reported in Table 2. Strikingly, the **both-correct** and **both-wrong** subsets yield comparable final distillation performance across different (student,teacher) pairs. For instance, on *Qwen2.5-0.5B-Instruct*, training on **both-wrong** improves over the pre-distillation baseline on all three benchmarks under both teachers, matching or surpassing the **both-correct** and random subsets. These observation indicates that:

[2]All Qwen3 models employed in our experiments default to non-thinking mode unless we specify otherwise.

Q1. *What does OPD learn?* §5.1 — DATA (CORRECTNESS)	**Q2.** *Where do its gains come from?* §5.2 — TOKEN (ENTROPY)	**Q3.** *Why does it sometimes degrade?* §5.3 — TOKEN (REWARD)
On-Policy Distillation learns reasoning ability rather than correctness: Training on subsets where the teachers *both succeed* or *both fail* yields comparable distillation performance, indicating that OPD transfers the *reasoning behavior* itself, not supervision correctness. *(Table 2)*	OPD aligns the student's *entropy shape* with the teacher's. Pre-OPD shape similarity (ρ, ForkOverlap) cleanly predicts the sign of the gain; the entropy-value gap ΔH does not. Gains come from closing the shape gap, so pairings already shape-aligned have no headroom. *(Table 3)*	Regression is driven by *destructive Q3 tokens*: at low-entropy positions on *correct* rollouts, the teacher pulls the student off tokens it had right. DEST:CORR >1 separates regressing from improving pairings; masking low-entropy positions causally removes the regression. *(Tables 4, 6)*

Figure 2: **Overview of §5.** Each column states one of the three diagnostic questions raised in §1, the corresponding subsection, and the key finding (with the table that supports it).

Setup	GSM8K	MATH-500	AMC
Qwen2.5-0.5B *(baseline)*	44.9	29.0	9.0
+ 7B [random]	45.7	32.8	9.0
+ 7B [both-correct]	**49.8**	**33.8**	8.2
+ 7B [both-wrong]	49.2	29.8	9.0
+ 14B [random]	48.5	32.8	8.2
+ 14B [both-correct]	48.1	30.0	8.2
+ 14B [both-wrong]	48.4	32.4	**9.7**
Qwen2.5-3B *(baseline)*	**85.8**	**63.8**	**33.6**
+ 7B [random]	84.4	60.2	27.6
+ 7B [both-correct]	83.7	63.0	27.6
+ 7B [both-wrong]	82.9	62.0	26.9
+ 14B [random]	84.0	62.0	29.1
+ 14B [both-correct]	83.2	61.6	26.1
+ 14B [both-wrong]	83.2	63.0	26.1

Table 2: Distillation performance on correctness-controlled subsets. Each student model is distilled from two teachers, *Qwen2.5-7B-Instruct* and *Qwen2.5-14B-Instruct*, on three subsets: *random* (10k prompts), *both-correct* (7k prompts), and *both-wrong* (7k prompts). The best result for each student model is **bolded**. AIME results are shown in Appendix B.

Insight 1. *On-Policy Distillation learns reasoning ability rather than correctness: distillation remains effective even when the teacher fails to produce a correct solution on any training problem.*

5.2 Token: Entropy Analysis

The knowledge-gap ablation shows that OPD's learning is independent of correctness, but what OPD actually learns remains unclear. Concurrent work (Li et al., 2026) identifies the progressive alignment of top-k tokens and the entropy gap, which appears to be a dynamic learning signal. However, the authors find that OPD can still fail even when the student exhibits a high top-k overlap ratio and shares a consistent thinking pattern with the teacher, and they attribute this failure to a lack of new knowledge. Yet the nature of this "knowledge" is left without deeper analysis.

In this section, we aim to uncover *what OPD actually learns* and trace *where its gains come from*. Inspired by the recent finding that RL reasoning gains are driven by a small set of high-entropy **fork tokens** at reasoning branch points (Wang et al., 2026), we argue that what matters for OPD is which positions are high-entropy (fork tokens) and whether the student and teacher agree on them. We further formalize this as the *Uncertainty Profile* (entropy shape) of reasoning trajectories.

Metrics. Let $H_t^p = H(p_t)$ and $H_t^q = H(q_t)$ denote the entropy of the student's and teacher's next-token distributions at trajectory position $t \in \{1, \ldots, T\}$. We define two metrics to quantify the alignment of the student's and teacher's uncertainty profiles along a single rollout:

- *Entropy Correlation.* The Pearson correlation between the student's and teacher's entropy sequences along the trajectory,

$$\rho = \text{Pearson}\big(\{H_t^p\}_{t=1}^T, \{H_t^q\}_{t=1}^T\big), \quad (4)$$

  measures whether the student's and teacher's entropies rise and fall in sync along the trajectory, capturing the *continuous* aspect of profile alignment.

- *Fork-Token Overlap.* Motivated by the notion of fork tokens, we identify the top-20% highest-entropy positions under the student and teacher distributions as their respective fork-token sets F^p and F^q, each of size $\lfloor 0.2\,T \rfloor$, capturing the positions each model deems most decisive:

$$\text{ForkOverlap} = \frac{|F^p \cap F^q|}{\lfloor 0.2\,T \rfloor}. \quad (5)$$

Student	Teacher	Correlation ρ			Fork Overlap (%)			Entropy Gap	Performance Δ (pp)		
		pre	post	Δ	pre	post	Δ	$\Delta(\Delta H)$	GSM8K	MATH-500	AMC
Qwen2.5-0.5B-It	7B-It	0.61	0.90	+0.29	73.6	82.1	+8.5	−0.050	+0.8	+3.8	0.0
	14B-It	0.56	0.97	+0.41	74.4	81.2	+6.8	+0.001	+3.6	+3.8	−0.7
	7B-GRPO	0.56	0.90	+0.34	70.5	66.2	−4.3	−0.080	+1.6	+0.4	+1.5
Qwen2.5-3B-It	7B-It	0.89	0.98	+0.09	83.6	85.8	+2.2	−0.031	−1.4	−3.6	−6.0
	14B-It	0.97	0.99	+0.02	82.4	83.5	+1.1	−0.004	−1.8	−1.8	−4.5
	7B-GRPO	0.78	0.96	+0.18	82.6	84.3	+1.7	−0.074	−2.0	−2.6	−2.3
Qwen3-1.7B	4B	0.96	0.94	−0.02	85.1	85.1	0.0	+0.018	−3.1	−2.6	−4.5
	4B-RL-Math	0.56	0.63	+0.07	76.4	82.1	+5.7	+0.055	−1.5	+0.5	+1.5

Table 3: **OPD transfers entropy shape, not entropy value.** Each row reports pre-OPD, post-OPD, and Δ for the shape metrics ρ and ForkOverlap; the entropy-gap change $\Delta(\Delta H)$; and accuracy deltas (pp) on GSM8K, MATH-500, and AMC. All teachers are Instruct versions; "7B-GRPO" denotes a Qwen2.5-7B-Instruct further trained with GRPO, and "4B-RL-Math" denotes a publicly released GRPO-tuned Qwen3-4B model.

This metric reflects whether the student and teacher agree on which positions are most uncertain—and thus most consequential for reasoning.

We additionally introduce a magnitude-based metric, the **Entropy Gap**, which measures the absolute difference between the student's and teacher's mean per-position entropies:

$$\Delta H = \left| \frac{1}{T}\sum_{t=1}^{T} H_t^p - \frac{1}{T}\sum_{t=1}^{T} H_t^q \right|. \quad (6)$$

This scalar summarizes the average gap in entropy value between the student and teacher, with no positional information.

Together, ρ and ForkOverlap characterize *entropy shape*, while ΔH captures *entropy value*.

Setup. We randomly sample 100 prompts from OpenR1-Math-220k as the probe set. For each (student, teacher) pairing we generate one student rollout per prompt at temperature 1.0, yielding 100 rollouts. All three metrics are computed *per rollout* and then averaged across the 100 rollouts.

We sweep two model families: *Qwen2.5-{0.5B, 3B}*-Instruct as students paired with *Qwen2.5-{7B, 14B}-Instruct* and a GRPO-tuned *Qwen2.5-7B-Instruct* as teachers (see recipe in Appendix A.2); and *Qwen3-1.7B* (Non-thinking) as student paired with *Qwen3-4B* and a public available *Qwen3-4B* GRPO variant[3] as teachers.

For each pairing, the *pre-OPD* setting uses rollouts from the vanilla student and the *post-OPD* setting uses rollouts from the corresponding OPD-trained checkpoint. For ρ and ForkOverlap, we report the pre-OPD value, the post-OPD value, and the OPD-induced change Δ; for the entropy gap, only $\Delta(\Delta H)$. Downstream performance is computed as the accuracy delta of the post-OPD student over its pre-distillation baseline on GSM8K, MATH-500, and AMC.

Results. Table 3 provides the two entropy shape metrics (ρ, ForkOverlap) before and after OPD, alongside the downstream accuracy deltas. For Qwen2.5-0.5B, both ρ and ForkOverlap start at low values ($\rho \in [0.56, 0.61]$, ForkOverlap $\in [70.5\%, 74.4\%]$), leaving substantial headroom; after OPD both metrics climb into the high range ($\rho \geq 0.90$), and every pairing yields downstream gains on GSM8K and MATH-500. This indicates that what OPD actually learn is the teacher's uncertainty profile. For Qwen2.5-3B the picture inverts: *pre-OPD* ρ and ForkOverlap are already high, the headroom for further alignment is limited, and downstream performance consistently regresses across all three benchmarks.

Generally, both shape metrics saturate in the *post-OPD* setting regardless of distillation results; what matters is the *pre-OPD* value, since a lower initial alignment with the teacher's entropy shape leaves more room for OPD to deliver gains. GRPO-tuned teachers fit the same pattern: their uncertainty profiles differ substantially from the base versions, lowering *pre-OPD* alignment and in some cases reversing the regression seen with the original teacher (e.g., Qwen3-1.7B regresses with the vanilla Qwen3-4B teacher but improves on MATH-500 and AMC with its GRPO-tuned counterpart). In contrast, the entropy gap $\Delta(\Delta H)$ is uncorrelated with downstream performance.

[3] https://huggingface.co/Keven16/Qwen3-4B-Non-Thinking-RL-Math-Step500

Insight 2. *OPD learns from the teacher's entropy* shape*, not its* value*; its gains come from shape alignment, leaving little headroom when student and teacher already match; GRPO rewrites the original model's uncertainty profile, which prior work describes as acquiring new "knowledge".*

5.3 Reward: Forgetting Analysis

The previous section only shows that OPD yields limited gains when the uncertainty profiles of the teacher and student are similar. However, it does not explain why OPD sometimes leads to performance degradation in practice (e.g., the results on Qwen2.5-3B-Instruct). We investigate this question from the perspective of catastrophic forgetting.

A per-token measure of steering strength. By the policy-gradient theorem, the token-level advantage of the OPD objective can be derived as (Yang et al., 2026):

$$r_t \triangleq \log q_t(\hat{y}_t) - \log p_t(\hat{y}_t), \tag{7}$$

which characterizes the strength with which OPD steers the student at each sampled token $\hat{y}_t$. We refer to r_t as the *per-token reward*. When $r_t > 0$, the teacher assigns higher probability to the student's sampled token than the student itself, and OPD pulls the student *toward* the teacher's preferred behavior. When $r_t < 0$, the teacher assigns lower probability to this token than the student does, and OPD instead pushes the student *away* from its own prediction, suppressing the token in future samples. The magnitude $|r_t|$ quantifies the strength of this push–pull effect.

Reward $\times$ Entropy quadrants. Prior analyses of catastrophic forgetting in supervised fine-tuning attribute the forgetting issue to gradient updates that force the model to fit low-probability targets at a positions where it is highly confident (Diao et al., 2026). Adapting this lens to OPD, we partition student-sampled tokens into four quadrants by crossing the sign of the per-token reward r_t with the per-token student entropy H_t^p, using an entropy threshold $H_{\text{th}} = 0.3$ nats (corresponding to a top-1 probability of ≈ 0.85). The resulting partition is shown in Table 5. Q3 is the direct analogue of the SFT forgetting trigger: the student is highly confident at position t, yet a negative reward from the teacher pushes it away from its own prediction.

Destructive vs. corrective. However, not every update from Q3 tokens is harmful. Sometimes the student is confidently wrong, in which case Q3 provides a useful corrective signal. We therefore further partition Q3 tokens by the correctness of the rollout they belong to:

- *Destructive Q3*: the student rollout reaches the *correct* final answer, yet at this position the teacher pulls the student off a confidently-correct token. Such updates erode behavior the student has already mastered.
- *Corrective Q3*: the student rollout reaches an *incorrect* final answer, and the teacher provides a correction at this position. Such updates are beneficial.

We report the destructive-to-corrective ratio Dest:Corr: a value below 1 indicates that the dominant update direction is corrective, while a value above 1 indicates that destructive updates dominate.

Setup. We reuse the probe set from §5.2. For each student$\times$teacher pairing, we sample one student rollout per prompt at temperature 1.0, yielding 100 rollouts in total. All sampled token positions across these 100 rollouts are pooled into a single population. Each token receives three labels: the binary entropy indicator $\mathbf{1}[H_t^p < H_{\text{th}}]$, the sign of r_t, and, for Q3 tokens, whether its parent rollout is correct.

The reported *mass* is the share of pooled tokens falling into the corresponding subset (destructive or corrective Q3); *mean* $|r|$ is the average reward magnitude over those same tokens. Pooling at the token level (rather than averaging per-rollout ratios) weighs each rollout by its length, which matches the actual gradient update strength of OPD.

Results. Table 4 presents the results. The DEST:CORR ratio cleanly separates the two regimes: every pairing in which OPD degrades the student is destructive-dominant (DEST:CORR $>$ 1), while every pairing in which OPD helps is corrective-dominant (DEST:CORR $<$ 1).

Causal validation via top-20% entropy masking. The above is correlational. To check that destructive Q3 force is *causally* responsible for the degradation, we run a targeted ablation: at training time the per-token KL loss is restricted to positions whose student-token entropy is in the *top-20%* of each rollout. Low-entropy positions, exactly where

Student	Teacher	Destructive Q3 mass (%)	Destructive Q3 mean $\lvert r\rvert$	Corrective Q3 mass (%)	Corrective Q3 mean $\lvert r\rvert$	Dest:Corr	Performance Δ (pp) GSM8K	MATH-500	AMC
Qwen2.5-0.5B-It	7B-It	5.5	0.186	15.9	0.186	0.34	+0.8	+3.8	0.0
	14B-It	9.0	0.147	22.1	0.166	0.36	+3.6	+3.8	−0.7
	7B-GRPO	4.4	0.303	11.2	0.286	0.41	+1.6	+0.4	+1.5
Qwen2.5-3B-It	7B-It	12.4	0.080	6.4	0.073	2.13	−1.4	−3.6	−6.0
	14B-It	21.4	0.057	9.0	0.070	1.95	−1.8	−1.8	−4.5
	7B-GRPO	5.1	0.203	4.3	0.159	1.52	−2.0	−2.6	−2.3

Table 4: **The destructive-to-corrective ratio predicts post-OPD regression.** For each student×teacher pairing, we decompose Q3 tokens by the correctness of their parent rollout: *destructive* Q3 (rollout is correct) and *corrective* Q3 (rollout is incorrect). For each group of Q3, we report the **mass** (share of all sampled tokens) and the **mean per-token reward magnitude** $|r_t|$, together with the Dest:Corr ratio and the post-OPD accuracy change relative to the vanilla student on GSM8K, MATH-500, and AMC (in percentage points).

	$H_t^p < H_{\text{th}}$ (*committed*)	$H_t^p \geq H_{\text{th}}$ (*forking*)
$r_t > 0$	Q1	Q2: useful distillation
$r_t < 0$	Q3: forced correction	Q4

Table 5: Four quadrants of student-sampled tokens, defined by the sign of the per-token reward r_t and the per-token student entropy H_t^p.

Method	GSM8K	MATH-500	AMC
Qwen2.5-3B (vanilla)	85.8	63.8	33.6
+ 14B, standard OPD	84.0	62.0	29.1
+ 14B, top-20% mask	83.8	64.4	32.8

Table 6: **Masking low-entropy tokens suppresses the degradation.**

destructive Q3 updates concentrate, are masked out and contribute no gradient. The shape-alignment signal at fork tokens (§5.2) is preserved while the forgetting mechanism is removed.

Applied to the Qwen2.5-3B × 14B-Inst pairing that regresses on GSM8K and AMC under standard OPD (Table 1), the entropy-masked variant recovers the bulk of the regression (Table 6): MATH-500 and AMC both return to near-vanilla levels, while GSM8K is essentially unchanged relative to standard OPD. Removing destructive force does not restore gains that did not exist under standard OPD, but it does suppress the larger losses on the other benchmarks.

Insight 3. *The regression observed under OPD stems from the teacher incorrectly suppressing the student's confident (low-entropy) tokens on* correct *rollouts. Masking out these low-entropy positions effectively prevents catastrophic forgetting in OPD.*

6 Conclusion

We presented a mechanistic analysis of on-policy distillation that explains both its gains and its failures. First, OPD transfers the teacher's *reasoning behavior* rather than its problem-level knowledge (§5.1). Second, gains come from aligning the student's *entropy shape* with the teacher's, so already shape-aligned pairings have no headroom to gain (§5.2). Third, the regression observed in such pairings is caused by *destructive* Q3 tokens and can be causally suppressed by masking low-entropy positions during training (§5.3).

Limitations

We highlight two limitations of this work. First, all experiments are conducted on math-reasoning task, and it remains unclear whether the capacity gap phenomenon and the identified mechanisms generalize to other domains such as code and open-ended settings. Second, due to limited computational resources, our study is restricted to relatively small student and teacher models; whether the same findings hold at at the larger scales remains an open question.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pages 21246–21263.

Mislav Balunovic, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. 2026. Matharena:

Evaluating llms on uncontaminated math competitions. *Advances in Neural Information Processing Systems*, 38.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28.

Dan Busbridge, Amitis Shidani, Floris Weers, Jason Ramapuram, Etai Littwin, and Russell Webb. 2025. Distillation scaling laws. In *International Conference on Machine Learning*, pages 5977–6045. PMLR.

Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. 2025. Retaining by doing: The role of on-policy data in mitigating forgetting. *arXiv preprint arXiv:2510.18874*.

Jang Hyun Cho and Bharath Hariharan. 2019. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4794–4802.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Muxi Diao, Lele Yang, Wuxuan Gong, Yutong Zhang, Zhonghao Yan, Yufei Han, Kongming Liang, Weiran Xu, and Zhanyu Ma. 2026. Entropy-adaptive fine-tuning: Resolving confident conflicts to mitigate forgetting. *arXiv preprint arXiv:2601.02151*.

Dongyi Ding, Tiannan Wang, Chenghao Zhu, Meiling Tao, Yuchen Eleanor Jiang, and Wangchunshu Zhou. 2025. Micota: Bridging the learnability gap with intermediate cot and teacher assistants. *arXiv preprint arXiv:2507.01887*.

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

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pages 32694–32717.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and 1 others. 2026. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*.

Tokio Kajitsuka, Ukyo Honda, and Sho Takase. 2026. Revisiting the capacity gap in chain-of-thought distillation from a practical perspective. *arXiv preprint arXiv:2604.08880*.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, and 1 others. 2024. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, and 1 others. 2026. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. 2025. Small models struggle to learn from strong reasoners. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25366–25394.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.

Kevin Lu and Thinking Machines Lab. 2025. On-policy distillation. *Thinking Machines Lab: Connectionism*. Https://thinkingmachines.ai/blog/on-policy-distillation.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Stéphane Ross and Drew Bagnell. 2010. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668. JMLR Workshop and Conference Proceedings.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. 2026. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin

Yang, Zhenru Zhang, and 1 others. 2026. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *Advances in Neural Information Processing Systems*, 38:115452–115486.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, and 1 others. 2026. Mimo-v2-flash technical report. *arXiv preprint arXiv:2601.02780*.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, and Radha Poovendran. 2025. Stronger models are not always stronger teachers for instruction tuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4392–4405, Albuquerque, New Mexico. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

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

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. 2024. Self-distillation bridges distribution gap in language model fine-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1028–1043, Bangkok, Thailand. Association for Computational Linguistics.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, and 1 others. 2026. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*.

Chen Zhang, Qiuchi Li, Dawei Song, Zheyu Ye, Yan Gao, and Yao Hu. 2025a. Towards the law of capacity gap in distilling language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22504–22528.

Chen Zhang, Yang Yang, Jiahao Liu, Jingang Wang, Yunsen Xian, Benyou Wang, and Dawei Song. 2023. Lifting the curse of capacity gap in distilling language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4535–4553, Toronto, Canada. Association for Computational Linguistics.

Hengyuan Zhang, Shiping Yang, Xiao Liang, Chenming Shang, Yuxuan Jiang, Chaofan Tao, Jing Xiong, Hayden Kwok-Hay So, Ruobing Xie, Angel X Chang, and 1 others. 2025b. Find your optimal teacher: Personalized data synthesis via router-guided multi-teacher distillation. *arXiv preprint arXiv:2510.10925*.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. 2026. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*.

Yuze Zhao, Jintao Huang, Jinghan Hu, Xingjun Wang, Yunlin Mao, Daoze Zhang, Zeyinzi Jiang, Zhikai Wu, Baole Ai, Ang Wang, Wenmeng Zhou, and Yingda Chen. 2024. Swift:a scalable lightweight infrastructure for fine-tuning. *Preprint*, arXiv:2408.05517.

A Training Details

A.1 On-Policy Distillation Training Details

All OPD experiments use the default hyperparameters listed in Table 7. We use the SWIFT framework (Zhao et al., 2024) and train on two nodes with 4×L40S GPUs each, using bf16 precision and gradient checkpointing. Student models are trained with DeepSpeed ZeRO-2, while teacher models use DeepSpeed ZeRO-3.

Hyper-parameter	Value
Training temperature	1.0
Rollout	1
Max prompt length	2048
Max response length	6144
LogProb Top-K	16
Top-K strategy	Student Top-K
Optimizer	AdamW
Learning rate	1e-5
Warmup Ratio	0.05
Effective batch size	32
Epoch	1

Table 7: **Default hyperparameters for OPD.**

A.2 GRPO Training Details

We train Qwen2.5-7B-Instruct with the Verl framework on the same 10,000 prompts used for the OPD training dataset. The hyperparameters are listed in Table 8.

Hyperparameter	Value
RL algorithm	GRPO
Training epochs	1
Train batch size	64
Rollout n	8
Maximum prompt length	1,024
Maximum response length	6,144
Learning rate	1×10^{-6}
Temperature	1.0
Top-p	1.0
KL loss type	low-variance KL
KL loss coefficient	1×10^{-3}
Loss aggregation	token-mean

Table 8: GRPO training hyperparameters for our Qwen2.5-7B-GRPO teacher.

B Additional Results: AIME

Table 9 reports pass@1 accuracy on AIME24/25 as a complementary evaluation for §4. And Table 10 presents the corresponding AIME results for the ablation in §5.1.

Setup	AIME24	AIME25
Qwen2.5-0.5B *(baseline)*	0.0	0.0
+ 7B	0.0	0.0
+ 14B	0.0	0.0
Qwen2.5-3B *(baseline)*	6.7	0.0
+ 7B	3.3	0.0
+ 14B	6.7	3.3
Qwen3-1.7B *(baseline)*	16.7	3.3
+ Qwen3-4B	3.3	6.7
+ Qwen3-8B	13.3	3.3
+ Qwen3-14B	10.0	6.7

Table 9: Pass@1 accuracy on AIME24 (n=30) and AIME25 (n=30) for the student × teacher pairs of Table 1.

Setup	AIME24	AIME25
Qwen2.5-0.5B *(baseline)*	0.0	0.0
+ 7B [random]	0.0	0.0
+ 7B [both-correct]	0.0	0.0
+ 7B [both-wrong]	0.0	0.0
+ 14B [random]	0.0	0.0
+ 14B [both-correct]	3.3	0.0
+ 14B [both-wrong]	0.0	0.0
Qwen2.5-3B *(baseline)*	6.7	0.0
+ 7B [random]	3.3	0.0
+ 7B [both-correct]	10.0	0.0
+ 7B [both-wrong]	10.0	3.3
+ 14B [random]	6.7	3.3
+ 14B [both-correct]	6.7	0.0
+ 14B [both-wrong]	6.7	0.0

Table 10: Pass@1 accuracy on AIME24 (n=30) and AIME25 (n=30) for the correctness-controlled runs of Table 2.