FILTER, THEN REWEIGHT: RETHINKING OPTIMIZATION GRANULARITY IN ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-Policy distillation (OPD) in large language models is shifting from full-trace KL supervision toward more selective training paradigms. Recent OPD methods increasingly focus on selecting which trajectories to learn from, which tokens are most informative, and which supervision signals are most reliable. Motivated by this trend, we rethink optimization granularity of OPD and propose 🔥 FiRe-OPD (Filter, then Reweight), which jointly adjusts supervision signals at both trajectory and token levels. In details, FiRe-OPD first filters trajectories to remove low-quality rollout samples, and then applies soft reweighting within the retained trajectories to emphasize informative tokens. Compared with hard token selection, FiRe-OPD leverages a soft-weighting mechanism to effectively mitigate information loss and enhance optimization stability, thereby achieving finer-grained OPD optimization. We validate the effectiveness of FiRe-OPD across strong-to-weak, single-teacher, and multi-teacher settings, and demonstrate its superiority over recent token-level OPD methods ((e.g., +6.25 on AIME 2024 in strong-to-weak, +18.81 on Miner in multi-teacher). Our code is available at `https://anonymous.4open.science/r/FiRe-OPD-7B28/README.md`.

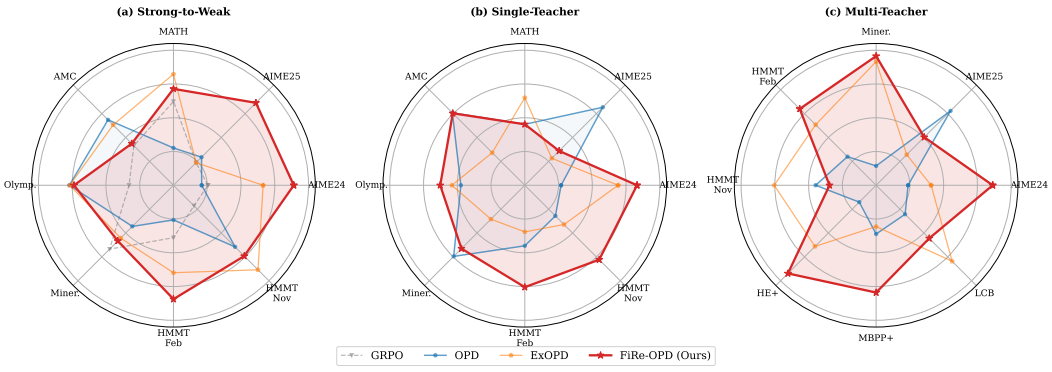

Figure 1: Performance comparison across three distillation scenarios. **FiRe-OPD** (red) achieves the most balanced and expansive coverage across all benchmarks

1 INTRODUCTION

On-policy distillation (OPD) has emerged as a compelling post-training paradigm for transferring reasoning capabilities from teacher models to smaller student models. Unlike supervised fine-tuning, OPD avoids the train-inference distribution mismatch by learning on student-generated trajectories, while providing denser token-level supervision than reinforcement learning's sparse outcome rewards (Zhu et al., 2026; Ye et al., 2026; Li et al., 2026b; Wu et al., 2026; Fu et al., 2026; Zheng et al., 2026; Jang et al., 2026; Song & Zheng, 2026). These advantages have made OPD a widely adopted approach in reasoning-intensive tasks.

However, standard OPD applies uniform full-trajectory KL supervision, which has inherent limitations in both optimization granularity and signal reliability. Not all trajectories and tokens carry

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.

equal learning value, and critical rollouts and informative tokens should be assigned greater importance during optimization. Recognizing this, selective optimization granularity distillation has become a growing trend in recent OPD research.

EOPD (Jin et al., 2026) identifies that high teacher entropy causes unstable learning signals and switches to forward KL at high-entropy token positions. TIP (Xu et al., 2026b) selects tokens based on student entropy and teacher-student divergence through hard filtering rules. ExOPD (Yang et al., 2026b) reinterprets OPD as KL-constrained RL and introduces a global reward scaling factor. Uni-OPD (Hou et al., 2026) addresses unreliable supervision through outcome-guided margin calibration at the trajectory level. But existing works suffer from two key limitations:

Limitation 1. *Granularity isolation.* Existing methods typically operate at either the trajectory or token level, focusing on only a single dimension of signal quality (e.g., teacher confidence or student state), without jointly modeling both granularities or exploiting their complementary in OPD.

Limitation 2. *Hard selection strategies on token.* Most existing token-level methods rely on hard selection strategies to remove certain tokens during OPD, which induces non-smooth optimization and permanently discards potentially useful and informative supervision signals, thereby weakening learning robustness. Table 1 provides a systematic and comprehensive comparison of existing OPD methods along these dimensions.

Table 1: Overview of OPD methods across granularities and techniques, and the scope of 🔥**FiRe-OPD**.

Method	Granularity		Technique		
	Traj.	Tok.	T-Conf.	S-Conf.	Soft-W.
OPD	✗	✗	✗	✗	✗
EOPD	✗	✓	✓	✗	✗
TIP	✗	✓	✗	✓	✗
ExOPD	✗	✗	✗	✗	✗
REOPOLD	✗	✓	✓	✗	✗
🔥**FiRe-OPD**	✓	✓	✓	✓	✓

In this work, we propose 🔥 **FiRe-OPD** (**Fi**lter, then **Re**weight), a unified framework that performs trajectory-level filtering and token-level importance weighting from a dual perspective of teacher confidence and student confusion. At the trajectory level, FiRe-OPD filters out rollouts where the teacher assigns low overall likelihood, indicating a large teacher-student distribution gap where the teacher's supervision is unreliable. At the token level, FiRe-OPD assigns continuous importance weights by multiplicatively combining teacher confidence and student confusion, concentrating learning on positions where the teacher provides reliable guidance and the student has genuine need. This soft weighting preserves gradient contributions from all positions proportional to their informativeness, enabling fine-grained, adaptive supervision that accounts for both *what the teacher can teach* and *what the student needs to learn*.

In summary, our contributions are 3-fold:

(i) We propose FiRe-OPD, a unified framework that jointly performs trajectory-level filtering and token-level soft reweighting, enabling fine-grained and selective OPD.

(ii) We show that optimization granularity is critical in OPD: hard filtering is more effective at the trajectory level, whereas soft token weighting surpasses hard token selection at the token level.

(iii) We show that the superiority of FiRe-OPD across strong-to-weak, single-teacher, and multi-teacher distillation settings on various benchmark.

2 Related Work

Knowledge distillation (KD) generally transfers knowledge from a stronger teacher to a smaller student model. Classical KD typically trains the student to match the teacher's output distribution, while sequence-level KD instead uses complete teacher-generated responses as supervision (Hinton et al., 2015; Kim & Rush, 2016). In the modern LLM era, KD has gradually evolved toward broader capability transfer such as reasoning and alignment (Gu et al., 2024; Ko et al., 2025; He et al., 2025a; Liu et al., 2024). However, most existing off-policy KD methods still rely on teacher-generated trajectories, often leading to exposure bias. These limitations motivate OPD, which directly supervises the student under its own generation distribution.

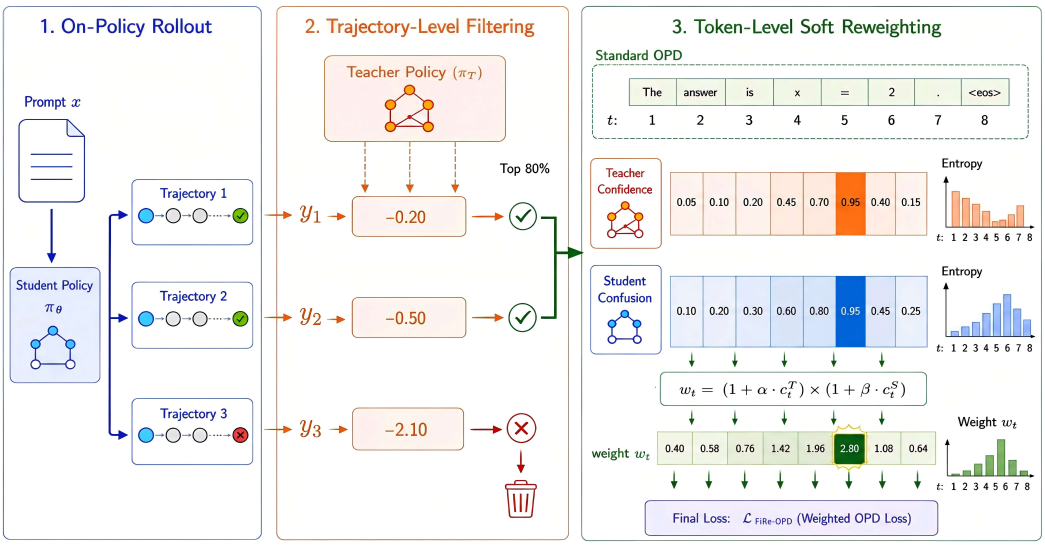

Figure 2: Overview of 🔥 **FiRe-OPD** that performs trajectory-level filtering and token-level importance weighting.

On-Policy Distillation. OPD has recently emerged as a highly effective and widely adopted paradigm for post-training. A number of prior studies have consistently shown that reverse-KL-style objectives and supervision on student-generated mistakes can substantially improve open-ended generation and reasoning tasks (Gu et al., 2024; Agarwal et al., 2024). A growing body of recent work further investigates how to effectively make OPD scalable, stable, and generalizable through reward extrapolation, entropy-aware objectives, reasoning-prefix acceleration, competence-aware curricula, divergence constraints, and rollout mixture distillation (Yang et al., 2026b; Jin et al., 2026; Zhang et al., 2026a; Luo et al., 2026; Hou et al., 2026). Meanwhile, OPD has also been successfully extended to self-distillation (Zhao et al., 2026; Xu et al., 2026a; Wang et al., 2026a; Kim et al., 2026; Zhang et al., 2026c; Yang et al., 2024), hybrid RL-distillation frameworks(Yan et al., 2026; Hübotter et al., 2026; Zhang et al., 2026d; Ding, 2026; Yang et al., 2026a; Zhang et al., 2026b), multimodal distillation(Li et al., 2026a; Cao et al., 2026; Chen et al., 2025; Bousselham et al., 2025), agentic settings(Wang et al., 2026b), and embodied learning(Zhong et al., 2026). In particular, recent token-selection methods generally attempt to reduce noisy supervision by discarding low-value tokens, but hard selection may often lose useful information and produce potentially brittle optimization signals. Our work directly addresses this limitation through a unified adaptive trajectory and token-level weighting scheme, which filters low-quality trajectories and softly modulates the token-level distillation intensity.

3 Methodology

3.1 Preliminaries

We first introduce the standard on-policy distillation (OPD) framework. Let π_θ denote the student model and π_T denote the teacher model. At each training iteration, the student generates rollouts from its current policy given a set of prompts $\{x_i\}$:

$$y \sim \pi_\theta(\cdot|x) \tag{1}$$

The teacher then provides token-level supervision on these student-generated trajectories. Standard OPD formulates this as a policy optimization problem using PPO-style clipped objectives, where the token-level advantage is defined as the teacher-student log-likelihood ratio:

$$a_t = \log \pi_T(y_t|x, y_{<t}) - \log \pi_{\theta_{\text{old}}}(y_t|x, y_{<t}) \tag{2}$$

Table 2: Strong-to-weak distillation results (Avg@8). Best results among OPD methods are in **bold**. Red/green denotes improvement/decline vs. OPD.

Method	AIME24	AIME25	MATH	AMC	Olymp.	Miner.	HMMT Feb	HMMT Nov	Avg
Strong-to-Weak: Qwen3-30B-A3B-Instruct → Qwen3-4B									
Student (Base)	21.67	22.50	83.65	67.19	51.80	39.48	12.50	7.08	38.23
Teacher	76.67	63.33	97.22	95.94	78.32	47.47	45.00	60.00	70.49
+ SFT	25.42	22.92	85.82	70.31	54.60	40.81	13.75	12.92	40.82
+ GRPO	55.00	48.33	93.20	93.06	68.69	43.73	29.17	35.42	58.33
+ OPD	54.58	48.75	91.25	**93.92**	70.62	43.01	28.33	39.17	58.70
+ ExOPD	58.75	48.33	**94.35**	93.75	70.61	43.38	30.83	**41.25**	60.16
+ TIP	59.58	49.58	92.19	93.60	70.66	**43.70**	29.58	40.00	59.86
+ REOPOLD	57.50	46.67	93.95	92.19	70.16	43.20	29.17	**41.25**	59.26
+ EOPD	52.92	49.17	93.40	92.81	**70.92**	42.97	27.08	39.17	58.56
+ 🔥 **FiRe-OPD (Ours)**	**60.83**	**52.92**	93.73	93.13	70.47	43.47	**32.08**	40.00	**60.83**
Δ vs OPD	+6.25	+4.17	+2.48	-0.79	-0.15	+0.46	+3.75	+0.83	+2.13

This advantage encourages the student to increase probability on tokens that the teacher assigns higher likelihood than the student's old policy. The policy gradient loss is:

$$\mathcal{L}_{\text{OPD}} = -\frac{1}{T}\sum_{t=1}^{T} \min\left(r_t a_t,\ \text{clip}(r_t, 1-\epsilon, 1+\epsilon)a_t\right). \tag{3}$$

where $r_t = \frac{\pi_\theta(y_t|x,y_{<t})}{\pi_{\theta_{\text{old}}}(y_t|x,y_{<t})}$ is the importance sampling ratio and the *clip* constrains r_t to $[1-\epsilon, 1+\epsilon]$ to prevent excessively large policy updates. We set $\epsilon = 0.2$. Standard OPD applies this objective uniformly across all trajectories and all token positions, treating every supervision signal equally.

3.2 🔥 FiRe-OPD

The standard OPD approach applies uniform supervision across all of the trajectories and token positions, which is suboptimal since the quality of distillation signals can vary significantly at both levels. As illustrated in Figure 2, FiRe-OPD effectively addresses this issue through two carefully designed complementary mechanisms: trajectory-level filtering and token-level soft reweighting.

Proposition 1. *What signal best reflects the importance of a trajectory?*

Some works use outcome correctness (Zheng et al., 2026; Hou et al., 2026) or reward scores to select trajectories. However, these approaches require external verifiers and do not directly reflect the teacher's supervision capability on a given path.

We observe that the teacher's log-probability on a student-generated trajectory typically reflects the underlying distributional alignment between the teacher and the student on that particular path. A relatively low teacher log-probability generally indicates a large distribution gap—the student's reasoning path diverges significantly from what the teacher would normally produce. In such cases, regardless of whether the trajectory is objectively correct, the teacher's token-level guidance along this path is often unreliable: the teacher is effectively being asked to supervise a reasoning style that it is unfamiliar with. Moreover, the supervision signal obtained on such trajectories tends to deviate excessively from the student's original distribution and is often unstable. Forcing distillation on these high-divergence trajectories can potentially introduce noisy or even contradictory gradients, ultimately leading to negative transfer rather than effective learning.

Based on this insight, we define the **trajectory-level importance score** as the teacher's normalized log-probability over a rollout $y = (y_1, \ldots, y_T)$ given prompt x:

$$s(y) = \frac{1}{T}\sum_{t=1}^{T} \log \pi_T(y_t|x, y_{<t}) \tag{4}$$

We rank all rollouts within a training batch by $s(y)$ and discard the bottom $p\%$ (we use $p = 20$ by default). Only the surviving trajectories proceed to the token-level optimization stage. This

Table 3: Single-teacher distillation results (Avg@8). Best results among OPD methods are in **bold**. Red/green denotes improvement/decline vs. OPD.

Method	AIME24	AIME25	MATH	AMC	Olymp.	Miner.	HMMT Feb	HMMT Nov	Avg
Single-Teacher: Qwen3-4B-Non-Thinking-RL-Math → Qwen3-4B									
Student (Base)	21.67	22.50	83.65	67.19	51.80	39.48	12.50	7.08	38.23
Teacher	56.25	46.67	93.40	91.56	68.06	47.52	30.83	35.83	58.77
+ OPD	57.92	**57.50**	94.69	**95.17**	47.66	**68.81**	32.50	35.42	61.21
+ ExOPD	60.42	54.58	**95.25**	93.44	47.84	67.28	32.08	35.83	60.84
+ 🔥 **FiRe-OPD (Ours)**	**61.25**	55.00	94.69	95.15	**48.07**	68.49	**33.75**	**37.50**	**61.74**
Δ vs OPD	+3.33	-2.50	+0.00	-0.02	+0.41	-0.32	+1.25	+2.08	+0.53

Table 4: Multi-teacher distillation results (Qwen3-4B-Non-Thinking-RL-Math + Qwen3-4B-Non-Thinking-RL-Code → Qwen3-4B-Non-Thinking). Best results are in **bold**.

Method	Math Reasoning (Avg@8)						Code Generation (pass@1)			
	AIME24	AIME25	Miner.	HMMT Feb	HMMT Nov	Avg	HE+	MBPP+	LCB	Avg
Student (Base)	21.67	22.50	39.48	12.50	7.08	20.65	79.90	64.60	17.57	54.02
Teacher	56.25	46.67	47.52	30.83	35.83	43.42	85.90	72.00	27.85	61.92
OPD	59.58	57.08	48.53	32.50	37.50	47.04	82.93	69.58	26.86	59.79
+ ExOPD	60.83	55.00	66.39	34.17	**38.75**	51.03	89.00	69.31	**29.28**	62.53
+ 🔥 **FiRe-OPD**	**64.17**	**55.83**	**67.34**	**35.00**	37.08	**51.88**	**92.70**	**71.69**	28.10	**64.16**
Δ vs OPD	+4.59	-1.25	+18.81	+2.50	-0.42	+4.84	+9.77	+2.11	+1.24	+4.37

filtering ensures that distillation occurs only on trajectories where the teacher can provide coherent supervision—paths that lie within the teacher's competence region, where its token-level signals are most likely to be meaningful and consistent.

Proposition 2. *What makes a token position informative for distillation?*

Existing works either focus on only a single signal—teacher entropy alone (Ko et al., 2026) or student entropy alone (Xu et al., 2026b)—or else apply hard truncation that discards tokens entirely below a fixed threshold. These approaches generally either miss one important dimension of signal quality or else irreversibly lose gradient information from positions that still carry some partial learning value. In contrast, we argue that a position is most informative for distillation when two conditions are jointly satisfied: the teacher is confident (providing reliable guidance) and the student is confused (indicating a genuine learning need). This naturally motivates a unified, soft weighting scheme that effectively integrates both signals simultaneously.

Based on this, we define the **token-level importance weight** using two complementary signals. Teacher confidence c_t^T measures how reliable the teacher's guidance is at position t:

$$c_t^T = 1 - \frac{H(\pi_T(\cdot|x, y_{<t}))}{\max_{t' \in \mathcal{B}} H(\pi_T(\cdot|x, y_{<t'}))} \tag{5}$$

where $H(\cdot)$ denotes the entropy of the output distribution and $\max_{t' \in \mathcal{B}}$ is the empirical maximum over all valid token positions in the current batch $\mathcal{B}$. Student confusion c_t^S measures how much the student needs guidance at position t:

$$c_t^S = \frac{H(\pi_\theta(\cdot|x, y_{<t}))}{\max_{t' \in \mathcal{B}} H(\pi_\theta(\cdot|x, y_{<t'}))} \tag{6}$$

The token-level importance weight combines both signals multiplicatively:

$$w_t = (1 + \alpha \cdot c_t^T) \times (1 + \beta \cdot c_t^S) \tag{7}$$

where $\alpha, \beta \geq 0$ are hyperparameters controlling the sensitivity to each respective factor (we use $\alpha = \beta = 1.0$ by default throughout all experiments). Our weighting scheme acts as a multiplicative

Table 5: Ablation study on the individual component contributions of 🔥FiRe-OPD (Avg@8, Strong-to-Weak setting). The best results are highlighted in **bold**.

Method	AIME24	AIME25	MATH	AMC	Olymp.	Miner.	HMMT Feb	HMMT Nov	Avg
OPD (Base)	54.58	48.75	91.25	**93.92**	70.62	43.01	28.33	39.17	58.70
🔥 **FiRe-OPD (Full)**	**60.83**	**52.92**	93.73	93.13	70.47	**43.47**	**32.08**	**40.00**	**60.83**
w/o Traj. Filter	56.92	49.88	93.54	91.19	70.41	43.12	28.33	38.42	58.99
w/o Teacher Conf.	58.33	52.08	**94.08**	90.31	**70.64**	42.46	28.75	37.92	59.32
w/o Student Conf.	59.08	49.87	93.51	91.34	70.33	43.00	31.17	37.92	59.53
Traj. Filter Only	54.58	51.67	93.85	91.88	69.96	42.88	30.42	39.17	59.30

soft AND gate: a token matters only when the teacher is confident and the student is confused. A purely additive combination would allow one factor to dominate the weight even when the other is uninformative, whereas the multiplicative form requires both factors to be large before the weight is substantially amplified. The weighted advantage for each token is then obtained by normalizing the raw weights within each trajectory to preserve gradient scale:

$$\tilde{a}_t = \frac{w_t}{\frac{1}{T}\sum_{t'=1}^{T} w_{t'}} \cdot a_t \tag{8}$$

The normalization ensures that the total gradient magnitude remains stable across different trajectories. The final policy gradient loss is:

$$\mathcal{L}_{\text{FiRe-OPD}} = -\frac{1}{T}\sum_{t=1}^{T} \min\big(r_t\tilde{a}_t,\ \text{clip}(r_t,\ 1-\epsilon,\ 1+\epsilon)\tilde{a}_t\big) \tag{9}$$

Overall, this design concentrates the learning effort on those positions where the teacher is confident yet the student still remains confused, while simultaneously still preserving the gradient contributions from all of the positions in proportion to their relative informativeness.

4 Experiment

4.1 Experimental Setup

Models. To demonstrate the generalizability of FiRe-OPD, we evaluate across three distillation scenarios: (i) *Strong-to-Weak*, where Qwen3-30B-A3B-Instruct serves as the teacher and Qwen3-4B-Non-Thinking (Yang et al., 2025) as the student, testing the ability to bridge large capacity gaps; (ii) *Single-Teacher*, where Qwen3-4B-Non-Thinking-RL-Math (Yang et al., 2026b) teaches Qwen3-4B-Non-Thinking, testing transfer efficiency between same models; and (iii) *Multi-Teacher*, where Qwen3-4B-Non-Thinking-RL-Math and Qwen3-4B-Non-Thinking-RL-Code (Yang et al., 2026b) jointly supervise Qwen3-4B, testing the ability to integrate heterogeneous domain expertise.

Training Data. For the strong-to-weak and single-teacher scenarios, we use the DeepMath-103K (He et al., 2025b) dataset filtered to difficulty level 6, following (Yang et al., 2026b). For the multi-teacher scenario, we use the multi-teacher training dataset from (Yang et al., 2026b), which combines mathematical and code-domain data.

Training Details. We train for 3 epochs (165 steps total) with a batch size of 1024, learning rate of 1×10^{-6}, and maximum response length of 16384. During rollout, we sample with temperature 1.0 and top-p = 1.0. For FiRe-OPD-specific hyperparameters, we set $\alpha = \beta = 1.0$ and trajectory filtering percentile $p = 20\%$. Training is conducted on 8×A100 80GB GPUs.

Evaluation. For mathematical reasoning, we evaluate on eight benchmarks spanning a range of difficulty levels: AIME 2024, AIME 2025, MATH-500 (Hendrycks et al., 2021), AMC 2023, OlympiadBench (He et al., 2024), MinervaMATH (Lewkowycz et al., 2022), HMMT 2025 Feb, and HMMT 2025 Nov (Balunovic et al., 2025). We sample 8 responses per problem with temperature 1.0 and report Avg@8 accuracy. For code generation, we evaluate on three widely-used

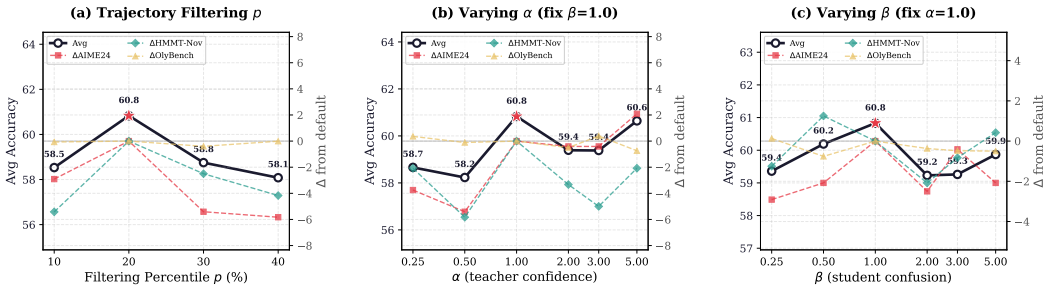

Figure 3: Hyperparameter sensitivity analysis. The solid black line (left axis) shows Avg accuracy across all benchmarks; dashed colored lines (right axis) show per-benchmark deviations (Δ) from the default setting. (a) Trajectory filtering percentile p exhibits a clear peak at p=20%. (b) Performance is robust for $\alpha \geq 1.0$ but degrades notably at small values. (c) β shows minimal sensitivity across the full range, confirming that student confusion weighting is robust to its scaling.

Table 6: Ablation study on the effectiveness of soft weighting versus hard truncation (Avg@8, Strong-to-Weak setting). Here, "Hard/Soft" specifically denotes the trajectory-level filtering and token-level weighting strategy, respectively. The best results are highlighted in **bold**.

Method	AIME24	AIME25	MATH	AMC	Olymp.	Miner.	HMMT Feb	HMMT Nov	Avg
🔥 **FiRe-OPD (Hard + Soft)**	**60.83**	**52.92**	93.73	**93.13**	70.47	43.47	**32.08**	**40.00**	**60.83**
Hard + Hard	57.92	46.25	**93.95**	90.62	68.53	**43.15**	31.67	33.75	58.23
Soft + Soft	57.92	48.75	93.75	90.94	**70.75**	**43.15**	28.33	35.83	58.68
Soft + Hard	55.42	51.25	93.35	89.38	68.86	43.01	30.42	36.67	58.55

benchmarks: HumanEval+ (Liu et al., 2023), MBPP+ (Liu et al., 2023), and LiveCodeBench (v6 only, February 2025–May 2025) (Jain et al., 2025), and report pass@1 accuracy.

Baselines. We compare against standard OPD and five recent improvements: ExOPD (Yang et al., 2026b), TIP (Xu et al., 2026b), REOPOLD (Ko et al., 2026), EOPD (Jin et al., 2026), and Uni-OPD (Hou et al., 2026). ExOPD uses the official open-source implementation; TIP, REOPOLD, and EOPD are reproduced by us. All methods are trained under the same data, model, and compute budget for fair comparison. We also report SFT and GRPO results as reference.

4.2 Main Results

Strong-to-Weak Distillation. Table 2 presents results for distilling from a 30B teacher to a 4B student. FiRe-OPD achieves the highest average accuracy of 60.83%, outperforming the strongest baseline ExOPD (60.16%) by 0.67 points and standard OPD (58.70%) by 2.13 points. The improvements are particularly pronounced on challenging competition-level benchmarks: +6.25 on AIME 2024, +4.17 on AIME 2025, +3.75 on HMMT Feb, and +2.48 on MATH-500. We also observe that FiRe-OPD substantially outperforms both SFT (which barely improves over the base model) and GRPO, confirming the advantage of dense teacher supervision combined with adaptive weighting. Compared to other token-level methods, FiRe-OPD consistently outperforms TIP (+0.97 avg), REOPOLD (+1.57 avg), and EOPD (+2.27 avg), demonstrating the effectiveness of our method.

Single-Teacher Distillation. Table 3 shows results where the teacher and student share the same architecture size, representing a minimal distribution gap scenario. FiRe-OPD achieves the highest average accuracy of 61.74%, improving over standard OPD (61.21%) by 0.53 points and ExOPD (60.84%) by 0.90 points, with notable gains on competition-level tasks such as AIME 2024 (+3.33) and HMMT Nov (+2.08). The consistent gains confirm that FiRe-OPD remains beneficial even when the teacher-student distribution gap is small.

Multi-Teacher Distillation. Table 4 presents results where two domain-specialized teachers (math and code) jointly supervise one student. FiRe-OPD achieves the best math reasoning average of 51.88% (+4.84 over OPD) and code generation average of 64.16% (+4.37 over OPD). The gains are substantial across both domains: +18.81 on MinervaMAT and +4.59 on AIME 2024 for math,

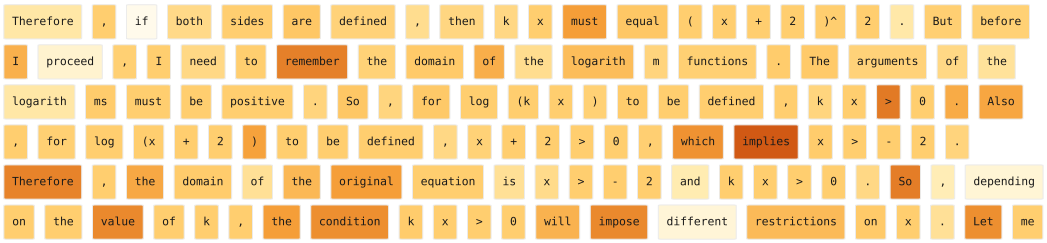

Figure 4: Case Study. Visualization of **FiRe-OPD**'s token-level weight allocation on a math reasoning trajectory.

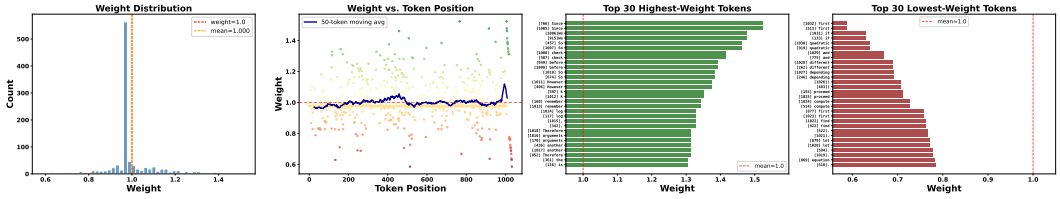

Figure 5: Statistical Analysis of Weight Allocation.

+9.77 on HumanEval+ and +2.11 on MBPP+ for code. Notably, FiRe-OPD enables the student to substantially surpass both teachers on code tasks (92.70 vs. 85.90 on HumanEval+), demonstrating effective knowledge integration beyond simple imitation.

Cross-Scenario Analysis. In the strong-to-weak setting, gains concentrate on competition-level benchmarks, confirming that trajectory filtering effectively removes low-quality rollouts that would otherwise corrupt learning on hard problems. Single-teacher distillation yields uniform but modest improvements given the smaller capacity gap, while the multi-teacher setting exhibits the most dramatic gains (+4.84 avg on math), as filtering and adaptive weighting naturally resolves conflicts between heterogeneous teachers. Overall, FiRe-OPD scales gracefully with distillation difficulty—whether from capacity gaps, task complexity, or supervision heterogeneity.

4.3 Ablation Studies

To gain deeper understanding of how each mechanism contributes to FiRe-OPD's effectiveness, we conduct comprehensive ablations analyzing the contribution of each component, the sensitivity to hyperparameters, and the effectiveness of soft weighting versus hard truncation. All ablations are performed in the Strong-to-Weak setting.

Component Ablation. Table 5 presents results when removing individual components. The full FiRe-OPD (60.83) significantly outperforms all ablated variants. Removing trajectory filtering causes the largest drop (-1.84), followed by teacher confidence (-1.51), while removing student confusion has the smallest impact (-1.30). This reveals an asymmetric role: trajectory-level filtering is the dominant contributor to FiRe-OPD's gains, while the two token-level signals provide complementary refinement, with teacher confidence playing a slightly larger role than student confusion. Trajectory filtering alone (59.30) already outperforms OPD (58.70), but combining it with token-level weighting yields further gains (+1.53), confirming that both granularities contribute complementarily. As an additional check, Appendix B further evaluates a verifier-aware filtering variant, which yields no gain over the default filtering rule.

Hyperparameter Sensitivity. Figure 3 visualizes sensitivity to the three hyperparameters. For filtering percentile p (Figure 3a), performance peaks at $p = 20\%$, with both under-filtering ($p = 10\%$: 58.53) and over-filtering ($p = 40\%$: 58.08) degrading results. For α (Figure 3b), performance is robust for $\alpha \geq 1.0$ but degrades at small values ($\alpha = 0.5$: 58.23), confirming that teacher confidence is essential for effective token-level weighting. For β (Figure 3c), performance shows minimal sensitivity across the full range, confirming that student confusion weighting is reasonably robust to its scaling. Complete per-benchmark results are provided in Tables 8 and 7. Additionally, we test difficulty-adaptive strategies that condition α, β, and the filtering percentile on the DeepMath diffi-

culty level in Appendix E, which do not outperform. Beyond hyperparameter choices, Appendix C further examines the stability of the empirical maximum in Eq. 6 under small batch sizes.

Soft Weighting vs. Hard Truncation. Table 6 compares four combinations of trajectory-level and token-level strategies (Hard=discrete filtering/selection, Soft=continuous weighting). FiRe-OPD's design (Hard trajectory filtering + Soft token weighting) achieves the best average of 60.83, outperforming Hard+Hard (58.23), Soft+Soft (58.68), and Soft+Hard (58.55). This validates two design choices: (1) At the trajectory level, hard filtering is superior to soft weighting, because low-quality trajectories should be completely removed rather than down-weighted, as even reduced gradients from unreliable paths can accumulate noise. (2) At the token level, soft weighting outperforms hard selection, since tokens exist on a continuum of informativeness, and preserving gradient contributions proportional to their value yields better optimization than binary keep-or-discard decisions.

4.4 Case Study: Token Weight Visualization.

To provide more intuitive understanding of how FiRe-OPD allocates learning effort, we visualize the token-level weights on a representative mathematical reasoning trajectory in Figure 4, where darker shading indicates higher weight. The highest weights are generally assigned to reasoning transition tokens such as "Therefore," "implies," and "So"—positions where the teacher confidently knows the next direction but the student remains uncertain. Conversely, the lowest-weight tokens are typically the locally predictable ones, including procedural words, routine reasoning words ("first," "if," "depending"), and formulaic punctuation, as both models are highly confident on them once the reasoning path is determined. Notably, the weighting is genuinely context-dependent: the same token "the" receives notably different weights depending on whether it introduces a critical reasoning conclusion or simply appears in a routine phrase.

Figure 5 further corroborates this pattern statistically through four complementary views of the learned token-level weight distribution. The first panel displays the overall weight histogram, which is sharply peaked around 1.0, confirming purely redistributive reweighting that preserves total gradient magnitude. The second panel presents a detailed positional analysis. It clearly shows that individual token weights are widely dispersed, while the 50-token moving average exhibits a distinct small peak near the end, where reasoning conclusions and final answers typically reside. The third and fourth panels list several representative tokens at the extremes of the weight spectrum: the highest-weight tokens are dominated by reasoning connectives ("Since," "So," "However," "Therefore") and metacognitive cues ("check," "remember"), while the lowest-weight tokens consist of procedural words ("proceed," "compute," "find") and formulaic punctuation. Together, these visualizations reveal that FiRe-OPD automatically identifies the distillation bottleneck as reasoning strategy selection—deciding *what to do next*—rather than computational execution, and concentrates learning effort on decision points where teacher guidance provides the greatest informational value. As an additional analysis, Appendix D examines a long 15,087-token trajectory and confirms that the weighting behavior does not decay over long horizons, with no systematic bias toward early tokens or degradation for deep tokens.

5 Conclusion

We propose FiRe-OPD, a dual-granularity framework for on-policy distillation that filters low-confidence trajectories and assigns continuous token-level weights based on teacher confidence and student confusion. Experiments across three distillation scenarios on math reasoning and code generation benchmarks demonstrate consistent improvements over standard OPD and recent baselines. Ablation studies reveal that teacher and student signals contribute asymmetrically across granularities, and that the two levels favor different selection strategies—hard filtering for trajectories and soft weighting for tokens.

AI use statement

In this work, we used generative AI tools for translation assistance, creating or modifying scientific figures or images, and editing the research paper to improve readability. We have not used generative AI tools for implementing methods, designing or providing feedback on research methodology or experiments, supporting qualitative and thematic data analysis, or interpreting results; the remaining

required disclosure tasks are not applicable to this work. All intellectual contributions, including research ideation, experimental design, data analysis, and the core arguments presented in this paper, were entirely our own. We have reviewed all AI-assisted work: translated content was manually checked by the authors, figure modifications were verified to preserve scientific accuracy, and text edits were reviewed for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The models, benchmarks, and training datasets used in this work are sourced from public repositories governed by licenses that permit their use in software and research. Our contributions fully adhere to the terms of these licenses. This study aims to improve on-policy distillation and poses no ethical concerns regarding privacy, harmful content generation, or potential misuse. The proposed method and experimental process do not involve any sensitive data or human subjects.

REPRODUCIBILITY STATEMENT

We provide comprehensive details to ensure reproducibility. We provide the complete codebase and detailed operating instructions. All experiments use publicly available datasets and open-source models. We specify model names, versions, dataset sources, and inference hyperparameters in Section 4.1. Training details, including batch size, learning rate, maximum response length, sampling temperature, and top-p, are also provided in Section 4.1. The mathematical formulations of our method, including the trajectory-level importance score, token-level importance weight, and final loss, are formally defined in Section 3.2. Complete hyperparameter settings (α, β, filtering percentile p) and ablation results are provided in Section 4.3 and the Appendix A. The experimental setups for the additional studies in Appendix B and Appendix E are also provided. We believe these details are sufficient to reproduce our results.

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 *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Mislav Balunovic, Jasper Dekoninck, Ivo Petrov, Nikola Jovanovic, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions, february 2025. *URL https://matharena.ai*, 8, 2025.

Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. Vold: Reasoning transfer from llms to vision-language models via on-policy distillation. *arXiv preprint arXiv:2510.23497*, 2025.

Di Cao, Dongjie Fu, Hai Yu, Siqi Zheng, Xu Tan, and Tao Jin. X-opd: Cross-modal on-policy distillation for capability alignment in speech llms. *arXiv preprint arXiv:2603.24596*, 2026.

Hansheng Chen, Kai Zhang, Hao Tan, Leonidas Guibas, Gordon Wetzstein, and Sai Bi. pi-flow: Policy-based few-step generation via imitation distillation. *arXiv preprint arXiv:2510.14974*, 2025.

Ken Ding. Hdpo: Hybrid distillation policy optimization via privileged self-distillation. *arXiv preprint arXiv:2603.23871*, 2026.

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

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

Changyi He, Yifu Ding, Jinyang Guo, Ruihao Gong, Haotong Qin, and Xianglong Liu. Da-kd: difficulty-aware knowledge distillation for efficient large language models. In *Forty-second International Conference on Machine Learning*, 2025a.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025b.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

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

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

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

Naman Jain, 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.

Ijun Jang, Jewon Yeom, Juan Yeo, Hyunggu Lim, and Taesup Kim. Stable on-policy distillation through adaptive target reformulation. *arXiv preprint arXiv:2601.07155*, 2026.

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

Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Why does self-distillation (sometimes) degrade the reasoning capability of llms? *arXiv preprint arXiv:2603.24472*, 2026.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. Distillm-2: A contrastive approach boosts the distillation of llms. In *International Conference on Machine Learning*, pp. 31044–31062. PMLR, 2025.

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.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

Jiaze Li, Hao Yin, Haoran Xu, Boshen Xu, Wenhui Tan, Zewen He, Jianzhong Ju, Zhenbo Luo, and Jian Luan. Video-opd: Efficient post-training of multimodal large language models for temporal video grounding via on-policy distillation. *arXiv preprint arXiv:2602.02994*, 2026a.

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

Jiaheng Liu, Chenchen Zhang, Jinyang Guo, Yuanxing Zhang, Haoran Que, Ken Deng, Zhiqi Bai, Jie Liu, Ge Zhang, Jiakai Wang, et al. Ddk: Distilling domain knowledge for efficient large language models. *Advances in Neural Information Processing Systems*, 37:98297–98319, 2024.

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.

Feng Luo, Yu-Neng Chuang, Guanchu Wang, Zicheng Xu, Xiaotian Han, Tianyi Zhang, and Vladimir Braverman. Demystifying opd: Length inflation and stabilization strategies for large language models. *arXiv preprint arXiv:2604.08527*, 2026.

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

Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, and Xiaoxin Chen. Skill-conditioned self-distillation for multi-turn llm agents. *arXiv preprint arXiv:2604.10674*, 2026a.

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. Tcod: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. *arXiv preprint arXiv:2604.24005*, 2026b.

Yecheng Wu, Song Han, and Hai Cai. Lightning opd: Efficient post-training for large reasoning models with offline on-policy distillation. *arXiv preprint arXiv:2604.13010*, 2026.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, and Zhipeng Wang. Paced: Distillation and on-policy self-distillation at the frontier of student competence. *arXiv preprint arXiv:2603.11178*, 2026a.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. Tip: Token importance in on-policy distillation. *arXiv preprint arXiv:2604.14084*, 2026b.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. *Advances in Neural Information Processing Systems*, 38:117157–117186, 2026.

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.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026a.

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.

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. 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)*, pp. 1028–1043, 2024.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

Dongxu Zhang, Zhichao Yang, Sepehr Janghorbani, Jun Han, Andrew Ressler II, Qian Qian, Gregory D Lyng, Sanjit Singh Batra, and Robert E Tillman. Fast and effective on-policy distillation from reasoning prefixes. *arXiv preprint arXiv:2602.15260*, 2026a.

Miaosen Zhang, Yishan Liu, Shuxia Lin, Xu Yang, Qi Dai, Chong Luo, Weihao Jiang, Peng Hou, Anxiang Zeng, Xin Geng, et al. Towards on-policy sft: Distribution discriminant theory and its applications in llm training. *arXiv preprint arXiv:2602.12222*, 2026b.

Xinsen Zhang, Zhenkai Ding, Tianjun Pan, Run Yang, Chun Kang, Xue Xiong, and Jingnan Gu. Opsdl: On-policy self-distillation for long-context language models. *arXiv preprint arXiv:2604.17535*, 2026c.

Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. Reinforcement-aware knowledge distillation for llm reasoning. *arXiv preprint arXiv:2602.22495*, 2026d.

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

Binbin Zheng, Xing Ma, Yiheng Liang, Jingqing Ruan, Xiaoliang Fu, Kepeng Lin, Benchang Zhu, Ke Zeng, and Xunliang Cai. Scope: Signal-calibrated on-policy distillation enhancement with dual-path adaptive weighting. *arXiv preprint arXiv:2604.10688*, 2026.

Zhide Zhong, Haodong Yan, Junfeng Li, Junjie He, Tianran Zhang, and Haoang Li. Vla-opd: Bridging offline sft and online rl for vision-language-action models via on-policy distillation. *arXiv preprint arXiv:2603.26666*, 2026.

Wenhong Zhu, Ruobing Xie, Rui Wang, and Pengfei Liu. Hybrid policy distillation for llms. *arXiv preprint arXiv:2604.20244*, 2026.

A Full Ablation Results

Sensitivity to α and β. Table 7 presents the full per-benchmark results for the entropy-aware weighting hyperparameters α and β in the strong-to-weak distillation setting. Overall, FiRe-OPD is reasonably robust to both parameters, with the average accuracy varying within a moderate range across all tested values. For α (teacher confidence scaling, with β fixed at 1.0), the performance variation is relatively larger than that of β, but the method remains competitive across a wide range: performance peaks at $\alpha = 1.0$ (60.83% avg) and stays close to the peak at $\alpha = 5.0$ (60.64%), indicating that moderately amplifying teacher confidence signals is beneficial while the method is not overly sensitive to this parameter. For β (student confusion scaling, with α fixed at 1.0), the overall variation is smaller than that of α, but the optimal value is more sharply defined: performance is best at $\beta = 1.0$, and deviations in either direction lead to degradation (e.g., HMMT-Feb drops from 32.08% to 29.17% at $\beta = 3.0$). This suggests that student confusion signals require more careful calibration than teacher confidence, because although β remains stable across a broad range, over-amplifying student confusion may cause the model to over-attend to positions where the learning signal is inherently noisy.

Sensitivity to Trajectory Filtering Percentile. Table 8 reports the effect of varying the trajectory-level filtering percentile p, which controls the fraction of lowest teacher-log-probability trajectories to discard. The optimal setting is $p = 20\%$, achieving 60.83% average accuracy. Lower filtering ($p = 10\%$) retains too many off-distribution trajectories that introduce noisy gradients, while aggressive filtering ($p = 30\%$ or $p = 40\%$) discards potentially useful training signals, particularly hurting performance on the most challenging benchmarks—AIME 2024 drops from 60.83% to 55.00% at $p = 40\%$, and HMMT-Feb drops from 32.08% to 26.25%. This confirms that a moderate filtering threshold strikes the best balance between removing harmful trajectories and preserving sufficient training diversity.

Table 7: Full ablation on α and β (Avg@8, Strong-to-Weak setting). Default: $\alpha = 1.0, \beta = 1.0$.

	Varying α (fix $\beta = 1.0$)						**Varying β (fix $\alpha = 1.0$)**					
Benchmark	α=0.25	α=0.5	α=1.0	α=2.0	α=3.0	α=5.0	β=0.25	β=0.5	β=1.0	β=2.0	β=3.0	β=5.0
AIME24	57.08	55.42	**60.83**	60.42	60.42	62.92	57.92	58.75	**60.83**	58.33	60.42	58.75
AIME25	47.50	48.75	52.92	49.17	51.25	**53.33**	48.33	**53.33**	52.92	50.00	48.75	50.00
MATH500	93.20	93.65	93.73	93.70	93.47	93.88	93.50	93.58	93.73	94.03	93.40	**94.27**
AMC2023	90.62	92.19	93.13	90.62	90.62	**93.75**	92.19	92.19	93.13	91.25	90.31	90.94
OlympiadBench	**70.83**	70.36	70.47	69.97	70.86	69.73	70.60	69.71	70.47	70.10	69.99	69.97
MinervaMAT	43.34	42.97	43.47	**44.12**	42.97	42.78	43.57	43.15	43.47	42.65	42.88	**44.21**
HMMT-Feb	28.75	28.33	**32.08**	30.42	30.42	30.83	30.00	29.58	**32.08**	29.58	29.17	30.42
HMMT-Nov	37.92	34.17	**40.00**	36.67	35.00	37.92	38.75	**41.25**	**40.00**	37.92	39.17	40.42
Avg	58.66	58.23	**60.83**	59.39	59.38	60.64	59.36	60.19	**60.83**	59.23	59.26	59.87

Table 8: Ablation on trajectory filtering percentile p (Avg@8). $p = 20\%$ is our default.

Benchmark	p=**10**	p=**20**	p=**30**	p=**40**
AIME24	57.92	**60.83**	55.42	55.00
AIME25	49.17	**52.92**	50.42	47.92
MATH500	93.60	93.73	93.45	**94.05**
AMC2023	91.25	93.13	90.31	91.88
OlympiadBench	70.40	**70.47**	70.07	70.46
MinervaMAT	42.97	43.47	43.24	**43.24**
HMMT-Feb	28.33	**32.08**	29.58	26.25
HMMT-Nov	34.58	**40.00**	37.50	35.83
Avg	58.53	**60.83**	58.75	58.08

B Verifier-Aware Filtering and Its Effect on Exploration

Motivation. Strict filtering by teacher log-probability may risk discarding correct but novel student reasoning paths. If the student discovers a valid solution that the teacher would not have produced,

its teacher log-probability can be extremely low, and the trajectory would be removed by the default FiRe-OPD filtering rule. To examine whether such correct-but-low-likelihood trajectories should be preserved, we introduce a sparse verifier signal into the filtering rule.

Experimental Setup. All experiments in this section are conducted under the strong-to-weak setting (Qwen3-30B-A3B-Instruct → Qwen3-4B) on the DeepMath-103K dataset (He et al., 2025b) filtered to difficulty level 6, following the default configuration in Section 4.1. Training hyperparameters are identical to the main experiments. Evaluation is performed on four competition-level benchmarks (AIME 2024, AIME 2025, HMMT 2025 Feb, HMMT 2025 Nov) with Avg@8 accuracy.

Verifier-Aware Filtering Rule. Let $s(y)$ be the teacher's average log-probability over a rollout y (Eq. 4), and let τ_p be the p-th percentile threshold used in the default filtering rule. We introduce a sparse verifier signal $v(y) \in \{0, 1\}$, where $v(y) = 1$ indicates that the final answer of y matches the ground truth. The modified retain condition is:

$$\text{Retain Condition:} \quad s(y) > \tau_p \quad \text{or} \quad v(y) = 1. \tag{10}$$

In other words, any trajectory that would normally be filtered out (bottom $p\%$ by teacher log-probability) is exempted from filtering if its final answer is correct. We use the default filtering percentile $p = 20\%$ throughout this experiment.

Results. As shown in Table 9, forcibly preserving correct but low-likelihood trajectories does not yield expected gains. The verifier-aware variant achieves an average of 44.00%, which is 2.46 points lower than the default FiRe-OPD (46.46%). The drop is consistent across all four benchmarks, ranging from –1.25 on HMMT25-Feb to –3.33 on HMMT25-Nov.

Table 9: Verifier-aware filtering vs. FiRe-OPD under the strong-to-weak setting (Avg@8). The verifier signal exempts correct trajectories from filtering.

Benchmark	Verifier Filtering	FiRe-OPD	Δ
AIME 2024	58.33	**60.83**	–2.50
AIME 2025	50.17	**52.92**	–2.75
HMMT25-Feb	30.83	**32.08**	–1.25
HMMT25-Nov	36.67	**40.00**	–3.33
Average	44.00	**46.46**	–2.46

Analysis. The negative result suggests that the default filtering mechanism already prevents penalizing gradients on correct trajectories. In standard OPD, a correct but low-teacher-likelihood trajectory would produce a large negative advantage (Eq. 2) and thus strong penalizing gradients. By filtering it out, FiRe-OPD avoids these gradients entirely, which is beneficial even if the trajectory is correct. When we exempt such trajectories via the verifier signal, they re-enter training with extremely weak teacher supervision—the teacher's token-level signals along these paths are unreliable, resulting in high-variance and noisy gradients. These noisy gradients can interfere with optimization and degrade overall performance, outweighing any benefit from preserving the correct final answer. Thus, moderate filtering serves as a denoising mechanism that protects the student from unreliable teacher supervision, rather than suppressing useful exploration.

Does filtering suppress exploration? A related concern is whether repeatedly filtering low-teacher-likelihood trajectories suppresses exploration and caps the student's ability ceiling. In standard OPD, if the student generates a novel path that the teacher assigns extremely low log-probability, Eq. 2 produces a large negative advantage, generating strong penalizing gradients. In FiRe-OPD, these high-divergence trajectories are directly filtered—mathematically, no gradient updates are applied to these divergent exploratory paths, avoiding the penalizing gradients that standard OPD would forcibly inject. Therefore, moderate filtering does not suppress exploration; it prevents over-penalization of exploratory paths and acts as a denoising mechanism.

C Batch Size Stability

The student confusion metric in Eq. 6 relies on an empirical maximum over the current batch, which may raise concerns about stability at small batch sizes. This is quantified with two extreme settings:

- *Extreme small batch (batch=4):* With 4 sequences per batch across 20 batches, the token weight maximum records a mean of 1.74 with a standard deviation of 0.09. The coefficient of variation (CoV) is 5.1%, well below the 10% stability threshold.
- *Practical large batch (batch=1024):* The empirical maximum shows even stricter stability, with a mean of 1.70, standard deviation of 0.09, and CoV of 5.0%.

These results confirm that the empirical maximum remains stable even under severely constrained batch capacity.

D Long-Trajectory Analysis

To examine whether soft weighting decays over long horizons, a dedicated experiment is conducted on a 15,087-token trajectory (close to the 16k maximum), recording per-token weights w_t:

- *No decay, no drift.* The mean weights across the first, middle, and last thirds are 0.992, 0.999, and 1.006, respectively. These values are nearly constant, with only a very mild upward trend that is negligible relative to the standard deviation of the per-token weights (0.095), indicating no bias toward early tokens and no decay for deep tokens.
- *Maximum weight near the end.* The maximum weight (1.703) occurs at position 13,662, corresponding to a reasoning conclusion—semantically driven, not positionally driven.
- *Conservative and localized weighting.* The vast majority of weights fall within 0.9–1.15.
- *Minimum weight at low-information tokens.* The minimum weight (0.621) occurs at position 14,979, corresponding to a closing punctuation—confirming weights are context-dependent, not position-dependent.

Table 10: Per-token weight statistics on a 15,087-token trajectory.

Metric	Value
Weight mean	1.000
Weight std	0.095
Weight max	1.703 (at position 13,662)
Weight min	0.621 (at position 14,979)
p90 / p10	1.133 / 0.953

Table 11: Mean weight across three equal segments.

Segment	Mean Weight
First third	0.992
Middle third	0.999
Last third	1.006

E Difficulty-Adaptive Distillation

Setup. This experiment tests whether sample-difficulty-dependent weighting and filtering can outperform the fixed hyperparameters used in FiRe-OPD. All experiments are conducted under the strong-to-weak setting (Qwen3-30B-A3B-Instruct $\rightarrow$ Qwen3-4B) on the DeepMath-103K dataset (He et al., 2025b), where each training sample is annotated with an integer difficulty level. Following (Yang et al., 2026b), the default setting uses difficulty level 6. For this experiment, we mix samples from level 5 and level 6 to obtain a difficulty-heterogeneous training set, so that the adaptive strategy can be conditioned on sample difficulty.

Adaptive Strategy. Let $d \in \{5, 6\}$ denote the difficulty level of a sample, and let $d_{\text{norm}} \in [0, 1]$ be its normalized difficulty defined as

$$d_{\text{norm}} = \frac{d - d_{\min}}{d_{\max} - d_{\min}}, \tag{11}$$

where $d_{\min} = 5$ and $d_{\max} = 6$ in this experiment. We then make the two weighting coefficients and the filtering percentile functions of d_{norm}:

$$\alpha(d) = 1.0 + 0.5 \cdot d_{\text{norm}}, \tag{12}$$

$$\beta(d) = 1.5 - 0.7 \cdot d_{\text{norm}}, \tag{13}$$

$$p(d) = \begin{cases} 25\%, & d = 5 \\ 15\%, & d = 6 \end{cases} \tag{14}$$

The rationale is as follows: harder samples (level 6) contain rarer and more critical teacher-confident tokens, so we enlarge α to emphasize teacher confidence and reduce the filtering percentile p to preserve scarce hard signals; easier samples (level 5) have more concentrated confusion points, so we enlarge β to emphasize student confusion and apply stronger filtering ($p = 25\%$) to remove low-quality rollouts. Evaluation is performed on four competition-level benchmarks (AIME 2024, AIME 2025, HMMT 2025 Feb, HMMT 2025 Nov) with Avg@8 accuracy.

Results. As shown in Table 12, the adaptive strategy does not achieve positive gains (44.17% vs. 46.46% average over fixed hyperparameters). Possible reasons include: (i) frequently switching optimization objectives across difficulty levels (filtering ratio $15\% \leftrightarrow 25\%$, and varying α, β) causes inconsistent gradient updates across batches; and (ii) FiRe-OPD is already robust to hyperparameter choices (as shown in Figure 3), leaving little room for adaptive improvement.

Table 12: Difficulty-adaptive distillation vs. fixed hyperparameters (Avg@8, strong-to-weak setting). The adaptive strategy conditions α, β, and the filtering percentile p on the DeepMath difficulty level $d \in \{5, 6\}$.

Benchmark	OPD	FiRe-OPD	Adaptive	Δ
AIME 2024	54.58	**60.83**	**60.83**	0.00
AIME 2025	48.75	**52.92**	48.33	–4.59
HMMT25-Feb	28.33	**32.08**	30.42	–1.66
HMMT25-Nov	39.17	**40.00**	37.08	–2.92
Average	42.71	**46.46**	44.17	–2.29