# ON-POLICY EVOLVING DISTILLATION: EVOLVING STUDENTS THROUGH TEACHER ADAPTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

On-Policy Distillation (OPD) aligns and compresses large language models (LLMs) by matching supervision to the student's on-policy data distribution. A teacher's standalone accuracy is an incomplete proxy for its value in distillation: in our 1.7B experiments, simple privileged information lowers the base model's answer accuracy, while students distilled from the context-conditioned teacher improve substantially. We propose **O**n-**P**olicy **E**volving **D**istillation (**OPED**), a bidirectional distillation paradigm that continuously evolves the teacher from student feedback while retaining a consistent source of guidance. OPED introduces two components: Student-Driven Teacher Evolution (SDTE), which constructs preference pairs exclusively from current-student samples and updates the teacher via direct preference optimization, and Global Privileged Information (GPI), a fixed teacher-only context that maintains a common set of guiding principles throughout evolution. Experiments on AIME2024, AIME2025, and HMMT25 show that OPED consistently improves student performance over static-teacher distillation, with gains of 2.1 and 5.3 points in macro avg@12 accuracy for 1.7B and 8B models, respectively. Updating the teacher outperforms alternating student DPO and distillation by 5.9 percentage points in the feedback-routing comparison. The evolved teacher raises early one-token continuation utility by 0.7 percentage points on frozen student states. Teacher evolution also improves student accuracy when the fixed teacher-only context is off-topic.

## 1 INTRODUCTION

Large language models (LLMs) are increasingly adapted to specialized applications, and knowledge distillation has become a central approach for transferring the capabilities of powerful teacher models into more efficient student models (Hinton et al., 2015; Kim & Rush, 2016). Conventional distillation creates a mismatch between teacher supervision and the student's policy distribution (Agarwal et al., 2024). On-Policy Distillation (OPD) addresses this mismatch by generating samples under the current student policy and learning from teacher supervision on those on-policy samples (Agarwal et al., 2024; Gu et al., 2024). Recent approaches transfer context-induced behavior through

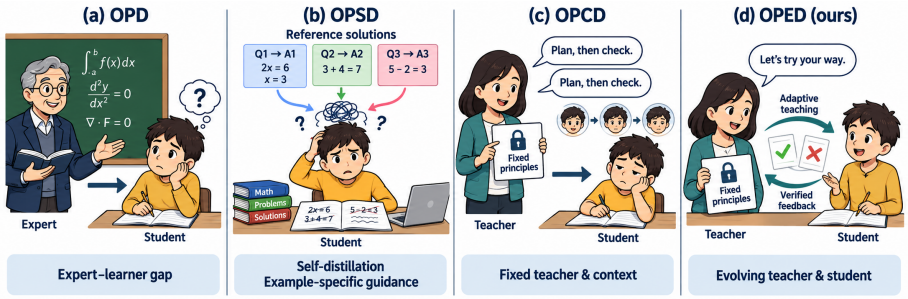


Figure 1: **Teaching paradigms for on-policy distillation.** (a) OPD: expert–learner gap. (b) OPSD: self-distillation with example-specific reference solutions. (c) OPCD: fixed-teacher baseline with fixed context. (d) OPED: teacher adaptation from verified student feedback under fixed GPI.



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.

privileged conditioning, including On-Policy Context Distillation (OPCD) (Ye et al., 2026) and On-Policy Self-Distillation (OPSD) (Zhao et al., 2026). In our Qwen3-1.7B experiments, adding simple procedural guidance to the base model changes standalone macro avg@12 from 35.9% to 33.3%. Students trained with the context-conditioned teacher reach 40.8% with OPCD and 42.9% with OPED. These results show that teaching value comes from the teacher's conditional supervision on student states, not from standalone accuracy alone. **Can the teacher evolve from student feedback to become more useful to that student?**

This observation motivates teacher adaptation. In static-teacher OPD, student feedback shapes only the student. The student continuously changes its output distribution, while the teacher parameters remain unchanged throughout training. Student exploration exposes reasoning attempts, errors, and capability gaps that can directly inform teacher updates.

Teacher adaptation also requires a consistent basis for guidance. OPED updates the teacher from verified comparisons involving student samples while preserving a fixed teacher-only context. The student's exploratory outputs on a separate feedback pool reveal current knowledge boundaries, failure modes, and learning trajectory. Comparing student-generated responses to the same problem supplies a natural signal for adapting the teacher when the responses differ in verified correctness. The teacher's global privileged information, consisting of broad task-level guidance, remains separate from the evolving parameters throughout training. The student drives teacher evolution while the teacher's guiding context stays fixed.

This formulation separates three notions that are often collapsed into a single measure of teacher quality: standalone solution accuracy, the usefulness of the teacher's action distribution on states visited by the current student, and the extent to which that signal can be absorbed through the chosen distillation objective. Teacher usefulness is therefore relational: it is jointly determined by the teacher, the student, and the on-policy states at which supervision is consumed. Student feedback is used to improve this relation by making teacher updates responsive to the student's evolving behavior.

We propose **O**n-**P**olicy **E**volving **D**istillation (OPED), a bidirectional distillation paradigm in which the student learns from the teacher while the teacher continuously evolves from student feedback. OPED introduces two complementary components. The first is Student-Driven Teacher Evolution (SDTE). At regular intervals, the current student samples responses from the feedback pool. Student-generated responses to each problem provide the candidates for adapting the teacher, while verified correctness supplies the preference signal. Both the preferred and dispreferred responses come from the current student, so the preference data reflects its current behavior. We then apply Direct Preference Optimization (DPO) (Rafailov et al., 2023) to update the teacher, allowing it to progressively adapt to information revealed through student exploration. Inspired by on-policy preference learning, using fresh student samples keeps the feedback tied to the behavior of the model that consumes the teacher's supervision. The second component is Global Privileged Information (GPI), which maintains continuity in teacher guidance during this evolution. We keep the teacher-only context fixed throughout the entire distillation process while allowing the teacher parameters to evolve. The student is no longer the endpoint of knowledge transfer; its on-policy behaviour becomes an additional source of supervision for improving the teacher. This creates an asymmetric feedback loop: the student learns through distillation, and the teacher adapts through preferences over the current student's responses. Figure 1 presents four teaching analogies: OPD depicts an expert teaching a puzzled student; OPSD depicts a student consulting a different worked answer for each problem, so privileged guidance varies across examples; our fixed-teacher OPCD baseline depicts a teacher repeatedly giving the same advice to an increasingly impatient student; and OPED depicts a teacher adapting to the student's progress. Concretely, GPI is the same teacher-only prompt/context at every teacher call used for student supervision, teacher adaptation, and teacher evaluation; the student never receives it directly. The prompt tokens and their ordering are fixed; SDTE modifies only the teacher's trainable parameters.

Our contributions are summarized as follows:

- We propose OPED, which extends static-teacher OPD into a bidirectional teacher-student co-evolution framework by enabling verified feedback involving current student samples to continuously influence teacher evolution.

- Student-driven teacher evolution with consistent guidance. We introduce SDTE and GPI, respectively, enabling feedback-driven teacher updates and preserving global privileged information as a stable source of guidance throughout teacher evolution.
- Empirical evaluation of teacher usefulness. On three mathematical reasoning benchmarks and two Qwen3 model sizes, OPED improves over our static-teacher OPCD baseline. Our analyses distinguish standalone teacher accuracy from downstream teaching value, compare teacher updates with alternating student DPO and distillation, and measure teacher utility on frozen student states.

The remainder of the paper reviews related work, presents OPED, reports experiments, analyzes teacher usefulness, and concludes with future directions.

## 2 Related Work

**Distillation on student trajectories.** Knowledge distillation transfers teacher predictions to students (Hinton et al., 2015), with sequence-level distillation extending this idea to autoregressive models (Kim & Rush, 2016). GKD supervises student-generated trajectories to address distribution mismatch (Agarwal et al., 2024); MiniLLM studies reverse-KL distillation (Gu et al., 2024); and DistiLLM combines skew-KL objectives with adaptive sample reuse (Ko et al., 2024). OPED additionally uses student responses to update the teacher supplying subsequent supervision.

**Privileged context and adaptive teachers.** OPSD uses privileged reference solutions to supervise student trajectories within a shared model (Zhao et al., 2026). OPCD transfers context-induced behavior through reverse-KL matching and supports separate-teacher and self-distillation configurations (Ye et al., 2026); we use its static-teacher configuration as our main baseline. Adaptive teaching is also studied by RLTs, which optimize explanations using student-understanding rewards (Cetin et al., 2025), and $\pi$-Distill, which jointly trains privileged and unprivileged roles within one model (Penaloza et al., 2026). CoPD interleaves verifiable-reward training with bidirectional distillation among experts (Gu et al., 2026). OPED instead updates a separate teacher through verified comparisons among the current student's responses, while retaining simple task-level instructions as fixed teacher-only context.

**Preference feedback and policy samples.** STaR iteratively trains on verified self-generated reasoning (Zelikman et al., 2022), whereas DPO learns from preferred and dispreferred responses (Rafailov et al., 2023). On-policy preference learning refreshes samples as the policy changes, with recent analysis examining evolving sampling coverage (Kim et al., 2026). OPED draws on this motivation but separates the sampling and optimized roles: student responses supply both members of each preference pair used to update the teacher. We compare this route with repeatedly alternating student DPO and distillation.

## 3 On-Policy Evolving Distillation

### 3.1 Problem Formulation and Global Privileged Information

Let $x \sim \mathcal{D}$ be a training problem, $\pi_\theta$ the student, and $\tau_\phi$ the teacher. The student conditions on $x$; the teacher additionally receives fixed privileged context $g$. We use separate adapters on the same base model, so the initial instructional asymmetry comes from teacher-only context. Students are evaluated without $g$.

Distillation uses $\tau_\phi(\cdot \mid g, x, \hat{y}_{<t})$ on student-generated prefixes. The teacher's contribution depends on how its conditional distributions guide the reasoning states the student actually visits. A prompt can change these distributions even when it lowers the teacher's own final-answer accuracy. OPED adapts this supervision through student feedback while retaining fixed privileged context.

GPI comprises three procedural rules: interpret the problem, plan the solution, and verify the result (Appendix C.1). The same task-level guidance, token ordering, and placement are used for teacher generation, student-rollout scoring, DPO, and teacher evaluation throughout training.

### 3.2 Student Learning from the Evolving Teacher

The student samples $\hat{y} \sim \pi_\theta(\cdot \mid x)$. Both models evaluate each student-generated prefix $\hat{y}_{<t}$, with the teacher additionally receiving $g$. Student learning minimizes

$$\mathcal{L}_{\mathrm{OPD}}(\theta; \phi, g) = \mathbb{E}_{x \sim \mathcal{D},\, \hat{y} \sim \pi_\theta(\cdot|x)} \left[ \frac{1}{|\hat{y}|} \sum_{t=1}^{|\hat{y}|} \min\{ D_{\mathrm{KL}}(\pi_\theta(\cdot \mid x, \hat{y}_{<t}) \| \tau_\phi(\cdot \mid g, x, \hat{y}_{<t})), c \} \right], \quad (1)$$

where $c$ is the pointwise KL cap. The teacher is treated as a fixed target within each student update, and gradients update only $\theta$. Supervision is therefore evaluated on states visited by the current student.

### 3.3 Student-Driven Teacher Evolution

At each evolution boundary, the current student generates candidate responses to problems from a separate feedback pool. A verifier orders responses to each problem by correctness; strict orderings yield preference pairs $(y^+, y^-)$. Both responses come from the current student. These comparisons capture successful reasoning and errors available under the learner's current policy, while verification supplies a concrete direction for adapting the teacher.

Following the motivation of on-policy preference learning (Kim et al., 2026), fresh student samples keep teacher adaptation responsive to the learner's evolving behavior. As the student improves, its responses reveal different reasoning attempts and failure modes. Refreshing the feedback lets the teacher update from these changing attempts.

The teacher and its fixed reference evaluate the likelihoods of these student-generated responses. With fixed reference teacher $\tau_{\mathrm{ref}}$, define

$$r_\phi(x, y; g) = \log \tau_\phi(y \mid g, x) - \log \tau_{\mathrm{ref}}(y \mid g, x). \quad (2)$$

The teacher objective is

$$\mathcal{L}_{\mathrm{SDTE}}(\phi) = -\mathbb{E}_{(x, y^+, y^-)} \left[ \log \sigma\big( \beta \left[ r_\phi(x, y^+; g) - r_\phi(x, y^-; g) \right] \big) \right], \quad (3)$$

where $\beta$ controls preference-update strength. SDTE updates only $\phi$. The evolved teacher then supplies targets for multiple subsequent student steps.

### 3.4 Alternating Updates

OPED alternates student-distillation blocks with teacher DPO. Student rollouts supply prefixes for token-level supervision, and fresh student responses supply verified preference pairs at evolution boundaries. Within each block, the teacher provides a fixed target while student parameters change. At the boundary, SDTE updates the teacher, and its revised distributions become the targets for subsequent student steps. This gives the two roles complementary objectives: the student matches token distributions, and the teacher learns from verified response preferences.

## 4 Experimental Evaluation

### 4.1 Experimental Protocol

We use Qwen3-1.7B and Qwen3-8B (Yang et al., 2025), with separate same-sized teacher and student LoRA adapters of rank 64 and scale 128 (Hu et al., 2022). Student training runs for 100 steps; evaluation uses step 90. Teacher updates occur periodically after steps 10 through 80. NuminaMath-CoT supplies 20,000 rollout problems, 750 teacher-feedback problems, 256 model-selection problems, and 256 diagnostic problems, with evaluation questions removed before allocation.

Each benchmark contains 30 problems with 12 stochastic responses per problem. Avg@12 is the mean single-response correctness rate; the macro average weights benchmarks equally. Evaluation uses temperature 1, top-$p = 1.0$, a 39,000-token generation horizon, and a 40,960-token context, with shared extraction, stopping, and failure-scoring rules. OPCD and OPED use reverse KL; OPSD uses forward KL. Five paired training seeds support the main comparison; paired bootstrap intervals resample seeds and problems. Optimization details appear in Appendix A.

### 4.2 Standalone Accuracy and Distillation Outcomes

Table 1 compares standalone Qwen3-1.7B with and without GPI against distilled students. Base + GPI is the initial teacher with procedural guidance and no parameter updates; students are evaluated without that context.

Table 1: Standalone model accuracy and downstream student accuracy for Qwen3-1.7B (avg@12, %). The context-conditioned base model is the initial teacher. OPCD and OPED rows report five-replicate student means.

| Model evaluated | AIME24 | AIME25 | HMMT25 | Mean |
|---|---|---|---|---|
| Base | 49.4 | 36.1 | 22.2 | 35.9 |
| Base + GPI | 45.0 | 33.9 | 21.1 | 33.3 |
| OPCD student | 53.6 | 40.9 | 27.7 | 40.8 |
| OPED student | **55.1** | **43.5** | **30.2** | **42.9** |

GPI lowers standalone macro avg@12 from 35.9% to 33.3%. In contrast, static-teacher OPCD produces a 40.8% student, and OPED further improves it to 42.9%. The teacher provides useful supervision through distributions on student-generated prefixes, whose teaching value differs from the teacher's own final-answer accuracy.

### 4.3 Repeated Comparison Across Model Scales

Table 2 compares matched runs with the same GPI, student training, model size, and evaluation protocol. OPCD fixes teacher parameters; OPED updates them through SDTE.

Table 2: Five-replicate mean avg@12 accuracy (%). Macro intervals are 95% simultaneous intervals for the two size-specific comparisons.

| Size | Method | AIME24 | AIME25 | HMMT25 | Mean |
|---|---|---|---|---|---|
| 1.7B | OPCD | 53.6 | 40.9 | 27.7 | 40.8 |
| | OPED | 55.1 | 43.5 | 30.2 | 42.9 |
| | Difference (pp) | +1.5 | +2.6 | +2.4 | +2.1 |
| | Macro CI | | [1.1, 2.8] | | |
| 8B | OPCD | 74.2 | 65.2 | 46.6 | 62.0 |
| | OPED | 78.6 | 70.6 | 52.5 | 67.2 |
| | Difference (pp) | +4.4 | +5.4 | +6.0 | +5.3 |
| | Macro CI | | [3.9, 7.2] | | |

OPED improves macro accuracy by 2.1 points at 1.7B and 5.3 points at 8B, with both simultaneous intervals excluding zero. Gains occur on every benchmark and are larger at 8B. Comparisons with additional training methods follow in Section 4.4.

### 4.4 Comparison with Additional Baselines

Table 3 compares OPED with the base model, SFT, GRPO, OPSD, and OPCD at both model sizes.

OPED achieves the highest accuracy on all three benchmarks at both sizes among the evaluated methods. The matched OPCD–OPED comparison measures the benefit of adding teacher evolution.

Table 3: Comparison with additional baselines on avg@12 accuracy (%) for Qwen3-1.7B and Qwen3-8B.

| | Qwen3-1.7B | | | | Qwen3-8B | | | |
|---|---|---|---|---|---|---|---|---|
| Method | AIME24 | AIME25 | HMMT25 | Mean | AIME24 | AIME25 | HMMT25 | Mean |
| Base | 49.4 | 36.1 | 22.2 | 35.9 | 70.6 | 65.0 | 43.3 | 59.6 |
| SFT | 37.5 | 30.8 | 19.2 | 29.2 | 69.2 | 60.3 | 36.1 | 55.2 |
| GRPO | 47.8 | 35.0 | 22.8 | 35.2 | 72.2 | 65.8 | 40.8 | 59.6 |
| OPSD | 49.2 | 37.5 | 26.4 | 37.7 | 72.2 | 67.8 | 43.9 | 61.3 |
| OPCD | 53.6 | 40.9 | 27.7 | 40.8 | 74.2 | 65.2 | 46.6 | 62.0 |
| OPED | **55.1** | **43.5** | **30.2** | **42.9** | **78.6** | **70.6** | **52.5** | **67.2** |

*Note.* Mean equally weights the three benchmarks; bold marks column maxima. OPCD and OPED reproduce the five-replicate means in Table 2.

### 4.5 Feedback Routing and Teacher-Update Schedule

Table 4 compares feedback routes at matched update boundaries: teacher DPO in OPED versus student DPO followed by distillation from an unchanged teacher.

Table 4: Feedback routing at 1.7B over five training replicates.

| Route | Mean (%) | OPED minus control |
|---|---|---|
| OPED | 42.9 | — |
| Alternating student DPO and distillation | 37.0 | +5.9 |
| Extra OPED | 40.7 | +2.2 |

Teacher-directed feedback outperforms alternating student DPO and distillation by 5.9 points. It changes targets used by later student steps, whereas direct student updates leave those targets fixed. Extra OPED, which adds teacher-evolution and student-distillation updates, reaches 40.7%, below OPED's 42.9%.

A sparse schedule with only one teacher update at step 50 reaches 40.2%, versus OPED's 42.9%. The tested periodic schedule therefore improves on the single-update alternative by 2.7 points, supporting continued adaptation as the student evolves. Together, the routing and schedule comparisons connect the destination of feedback to its timing: updating the teacher affects subsequent supervision, and refreshing that teacher periodically keeps these targets responsive to the learner.

### 4.6 Does Current-Student Feedback Have Distinct Value?

We compare current-student, initial-student, and teacher sampling for teacher DPO. Each variant constructs both responses in a preference pair from the designated source; standard OPED uses the current student. Initial-student sampling anchors feedback to the learner's starting policy, while teacher sampling reflects the teacher's own response distribution. Current-student sampling tracks the policy that will consume the updated supervision. All three variants use a 1.7B teacher and student. We also test OPD from a fixed 8B teacher to the 1.7B student.

Table 5: Feedback-source and teacher-capacity comparisons for a 1.7B student (avg@12, %). OPED variants use a 1.7B teacher and pairs drawn entirely from the indicated source; OPD uses a fixed 8B teacher.

| Teacher / feedback configuration | AIME24 | AIME25 | HMMT25 | Mean |
|---|---|---|---|---|
| Current student (standard OPED) | **55.1** | **43.5** | **30.2** | **42.9** |
| Initial student (ablation) | 53.3 | 40.0 | 25.6 | 39.6 |
| Teacher (ablation) | 52.2 | 41.3 | 24.7 | 39.4 |
| Static 8B teacher (OPD) | 54.0 | 39.4 | 25.2 | 39.5 |

Current-student feedback yields the highest accuracy on all three benchmarks (Table 5). Compared with initial-student sampling, standard OPED improves AIME24, AIME25, and HMMT25 by 1.8, 3.5, and 4.6 percentage points, respectively, with a macro gain of 3.3 points. The corresponding gains over teacher sampling are 2.9, 2.2, and 5.5 points, with a macro gain of 3.5 points. Current-student samples track the learner's changing policy and provide the strongest feedback source for adapting the teacher to the learner that consumes its supervision.

**Teacher adaptation versus capacity.** OPD from the fixed 8B teacher yields a macro accuracy of 39.5%, below OPED's 42.9% with a 1.7B teacher. The smaller, adapting teacher improves student accuracy by 1.1, 4.1, and 5.0 percentage points on AIME24, AIME25, and HMMT25, respectively, with a macro advantage of 3.4 points. This comparison favors the smaller, adapting teacher and supports tracking the student's evolving behavior.

### 4.7 Evolution Under Different Fixed GPI Contexts

Table 6 compares procedural guidance, off-topic prose, and a reproducible 120-token random block. Each context remains byte-identical within a run; only teacher parameters evolve.

Table 6: Context-by-evolution comparison at 1.7B over five training replicates.

| Fixed teacher-only context | Static | Evolving | Gain (pp) | 95% CI |
|---|---|---|---|---|
| Off-topic prose | 36.3 | 38.0 | $+1.7$ | $[0.7, 2.7]$ |
| Random-token block | 36.5 | 36.9 | $+0.4$ | $[-0.4, 1.2]$ |
| Procedural rules | 40.8 | 42.9 | $+2.1$ | $[1.1, 2.8]$ |

Procedural rules give the strongest students with both static and evolving teachers. Evolution also helps under off-topic prose, and random context produces a smaller, statistically unresolved gain. Useful initial guidance and teacher adaptation both contribute to downstream performance. Removing GPI from a shared 10-step checkpoint weakens the student in the same-seed comparison (Appendix C), supporting continued access to the original guidance during evolution.

### 4.8 Teacher Usefulness on Frozen Student States

Let $s_t = (x, y_{<t}, t)$ be a state generated by frozen student $\pi$, and let $\tau$ denote a teacher with GPI. One-token continuation utility is

$$u_t(\tau; \pi, s_t) = \sum_a [\tau(a \mid s_t) - \pi(a \mid s_t)]\, Q_t^\pi(s_t, a), \tag{4}$$

where $Q_t^\pi(s_t, a)$ is expected final verification reward after action $a$ and continuation with the same student. This compares teacher-token and student-token interventions while fixing the prefix and continuation policy. We use 256 disjoint problems, two student rollouts each, one prespecified early and one late position, and 16 fresh token-and-continuation repetitions per state.

Table 7: One-token continuation utility in percentage points of final reward relative to the student-token control.

| Teacher intervention | Early positions | Late positions |
|---|---|---|
| Before DPO | $+0.4$ | $+0.1$ |
| After correct-label DPO | $+1.1$ | $+0.0$ |
| After shuffled-label DPO | $+0.3$ | $+0.1$ |
| Correct-label after minus before | $+0.7$ | $-0.1$ |
| 95% CI | $[0.1, 1.3]$ | $[-0.4, 0.2]$ |
| Correct-label after minus shuffled | $+0.9$ | $-0.1$ |
| Adjusted early CI | $[0.2, 1.5]$ | — |

Table 7 shows that correct-label DPO improves early utility by 0.7 points over the initial teacher and 0.9 over shuffled-label DPO, with intervals excluding zero. The late-position difference from

the initial teacher is $-0.1$ points, with an interval spanning zero. Teacher adaptation improves local guidance at the tested early positions, independently of subsequent student training.

## 5 Analysis of Teacher Usefulness

**Guidance is useful in relation to the student.** Standalone teacher accuracy, teacher utility on student states, and downstream student improvement measure different behaviors. Standalone evaluation samples complete solutions from the teacher; the frozen-state intervention tests a teacher action followed by continuation under the same student; downstream evaluation measures what the student absorbs through training. Simple GPI lowers the teacher's own solution accuracy, yet the conditioned teacher supports effective distillation. SDTE further increases its early utility on frozen student states. Reading these measurements together connects conditional guidance to learning outcomes and motivates evaluating teachers by the supervision they provide to a particular learner.

**Consistent guidance and adaptive targets.** GPI provides consistent guidance, while fresh student samples make teacher updates responsive to current reasoning attempts. Current-student feedback outperforms initial-student and teacher sampling, and OPED with a 1.7B teacher outperforms OPD from a fixed 8B teacher (Table 5).

Teacher DPO changes a target reused across later student steps, helping explain the advantage over direct student DPO in the routing comparison. Appendix B illustrates this persistence with the local update $p_{n+1} = (1-\eta)p_n + \eta q$, with student distribution $p_n$, fixed teacher target $q$, and step size $0 < \eta < 1$. For a perturbation $\delta$ over $h$ steps, a direct student update contributes $(1-\eta)^h\delta$, while a persistent target perturbation contributes $[1-(1-\eta)^h]\delta$. This local model explains how a teacher update continues influencing the student within a distillation block. Periodic updates refresh that reusable target as the student's policy changes.

## 6 Conclusion

We studied how to improve a teacher for on-policy distillation when standalone solution accuracy diverges from teaching value. Simple privileged guidance lowers standalone Qwen3-1.7B accuracy in our comparison, yet a student distilled from that context-conditioned teacher improves substantially. OPED builds on this observation by periodically adapting a separate teacher with verified preference pairs constructed entirely from current-student responses. GPI retains a common instructional context throughout these updates.

Across AIME2024, AIME2025, and HMMT25, OPED improves macro avg@12 over our static-teacher OPCD baseline by 2.1 percentage points at 1.7B and 5.3 points at 8B. It also outperforms alternating student DPO and distillation in the feedback-routing comparison. Frozen-state interventions show an increase in early one-token teacher utility after correct-label DPO. Together, these results establish student-relative teacher adaptation as an effective mechanism for on-policy distillation.

These experiments establish OPED on competition mathematics across two Qwen3 model sizes, with teacher evolution driven by verified feedback pools and fixed task-level GPI. The framework opens direct extensions to adaptive update schedules, automatic construction of privileged guidance, and broader task families where student feedback can improve the teacher that supplies future on-policy supervision.

## AI Assistance

No AI assistants were used in the research or writing of this paper.

## REFERENCES


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

Edoardo Cetin, Tianyu Zhao, and Yujin Tang. Reinforcement learning teachers of test time scaling. *arXiv preprint arXiv:2506.08388*, 2025. URL https://arxiv.org/abs/2506.08388.

Naibin Gu, Chenxu Yang, Qingyi Si, Chuanyu Qin, Dingyu Yao, Peng Fu, Zheng Lin, Weiping Wang, Nan Duan, and Jiaqi Wang. Co-evolving policy distillation. *arXiv preprint arXiv:2604.27083*, 2026. URL https://arxiv.org/abs/2604.27083.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2306.08543.

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Juno Kim, Jihun Yun, Jason D. Lee, and Kwang-Sung Jun. Coverage improvement and fast convergence of on-policy preference learning. *arXiv preprint arXiv:2601.08421*, 2026. URL https://arxiv.org/abs/2601.08421.

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. Association for Computational Linguistics, 2016. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024. URL https://arxiv.org/abs/2402.03898.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. *arXiv preprint arXiv:2602.04942*, 2026. URL https://arxiv.org/abs/2602.04942.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023. URL https://arxiv.org/abs/2305.18290.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

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

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. *arXiv preprint arXiv:2203.14465*, 2022. URL https://arxiv.org/abs/2203.14465.

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. URL https://arxiv.org/abs/2601.18734.

# A Additional Training Details

Student optimization uses learning rate $5 \times 10^{-6}$, effective batch size 32, gradient-norm cap 0.1, and zero adapter dropout. Teacher DPO uses learning rate $10^{-6}$, batch size eight preference pairs, $\beta = 0.2$, and the initial teacher as its fixed reference. The pointwise KL cap is 0.1. The teacher-evolution pool supplies training feedback and is distinct from the model-selection and diagnostic pools. Five paired seeds are the independent training replicates for the main OPCD–OPED comparison; paired bootstrap intervals resample both seeds and evaluation problems. Other analyses specify their repetition counts separately.

Students are evaluated without GPI; standalone teachers retain their specified context. The direct-feedback baseline repeats student DPO at the corresponding evolution boundaries and resumes distillation from the unchanged teacher.

## A.1 Student Sampling and Preference-Pair Construction

**Student-only sampling.** At each teacher-update boundary, we freeze the current student checkpoint and sample eight responses for each problem in the fixed 750-problem teacher-evolution pool, yielding 6,000 responses per round. All responses used to construct preference pairs are generated by the student; the teacher does not generate candidate responses for this procedure. The student receives only the problem and the instruction "Please reason step by step, and put your final answer within `\boxed{}`." No privileged information or reference solution is included in the student's sampling prompt.

Sampling uses temperature 1.0, top-$p$ 0.95, and no top-$k$ restriction (`top_k=-1`). The maximum completion length is 8,192 tokens, and the maximum total context length is 12,288 tokens. We apply the Qwen3 chat template with `enable_thinking=False`. This setting disables the explicit thinking-mode template; it does not prohibit step-by-step reasoning in the generated response. These settings apply to feedback collection and are distinct from the final benchmark evaluation protocol.

**Verification and mixed-success filtering.** Each sampled response receives a binary correctness label based on verification of its extracted final answer against the reference answer. We extract the last complete `\boxed{}` answer. A response without an extractable answer is labeled incorrect. Reaching the generation-length limit does not, by itself, determine correctness: a length-limited response is verified using the same answer-extraction and correctness procedure as any other response. There is no additional filtering based on response length.

For problem $x$, let $c_x$ be the number of correct responses among the eight samples. We retain a problem for preference construction if and only if $1 \leq c_x \leq 7$. Problems with eight correct responses or eight incorrect responses contribute no preference pairs, because they do not provide a strict correct-versus-incorrect comparison. They are not resampled solely to obtain a mixed-success outcome. This filtering is repeated using fresh samples from the current student at every teacher-update boundary.

**Pair construction and per-problem balancing.** For each retained problem, let $\mathcal{Y}_x^+$ and $\mathcal{Y}_x^-$ denote its correct and incorrect sampled responses. The candidate preference pairs are the Cartesian product

$$\mathcal{P}_x = \mathcal{Y}_x^+ \times \mathcal{Y}_x^-, \qquad |\mathcal{P}_x| = c_x(8 - c_x).$$

Each pair contains a correct student response as the chosen response $y^+$ and an incorrect student response as the rejected response $y^-$. We randomly shuffle the candidate pairs and retain at most four pairs per problem, without replacement at the level of sampled-response combinations. The shuffle is reproducible and is determined by the round-specific seed and a stable hash of the problem identifier.

Since $c_x(8 - c_x) \geq 7$ for every retained problem, each mixed-success problem contributes exactly four pairs when all eight samples are available. This cap gives every retained problem the same pair count, rather than allowing problems with more correct–incorrect combinations to dominate the DPO dataset. We do not deduplicate responses by their text. Consequently, identical strings generated in separate sampling records may remain in the dataset, and an individual response may

participate in more than one selected pair. We do not rank or select pairs by response length or model likelihood.

**Problem-level training and validation split.** At each round, we split the retained mixed-success problems into approximately 80% training problems and 20% validation problems. The split operates on unique problem identifiers, not on individual responses or preference pairs. All selected pairs from a problem are assigned to the same subset, ensuring that no problem contributes pairs to both training and validation within that round. The validation-set size is obtained by rounding 20% of the retained problem count; the remaining problems form the training set. All constructed training pairs are used for the teacher's DPO update, and all constructed validation pairs are retained for DPO validation.

The problem split is recomputed at each round using a round-specific seed. Therefore, a problem assigned to validation in one round may have appeared in the training subset of another round. This validation subset provides within-round DPO diagnostics and should not be interpreted as a globally held-out generalization set. It is separate from the model-selection and diagnostic pools described in the main experimental protocol.

**Conditioning during teacher DPO.** Although both responses in each preference pair originate from the unprivileged student, the teacher scores these responses conditioned on the problem and the fixed privileged context. The reference answer is used for verification and preference labeling, not as additional text in the student's sampling prompt. Thus, student sampling supplies the candidate behaviors, while verified correctness supplies the preference labels.

## B A Local Illustration of Target Reuse

**Teacher updates change a target used repeatedly by the student.** The routing comparison repeatedly alternates preference learning and distillation in both conditions. Student DPO changes the learner while leaving its future distillation target fixed; teacher DPO changes the target used across subsequent student updates. As a local illustration, consider $p_{n+1} = (1-\eta)p_n + \eta q$, with fixed $q$ and $0 < \eta < 1$. After $h$ distillation steps, a direct student perturbation contributes $(1-\eta)^h\delta$, while a persistent target perturbation contributes $[1-(1-\eta)^h]\delta$. Applied to one interval between preference updates, this model illustrates how an updated teacher continues influencing the student. The routing experiment evaluates this repeated alternating process and motivates viewing the teacher as a reusable target through which verified feedback influences multiple student updates.

## C Retaining GPI During Teacher Evolution

Table 8 compares two same-seed Qwen3-1.7B runs from the same 10-step teacher checkpoint. Both use procedural GPI for the first 10 steps; one retains it thereafter and the other removes it. Retaining GPI yields 42.2% macro accuracy, versus 40.9% after removal.

Table 8: Downstream student avg@12 accuracy (%) in the same-seed Qwen3-1.7B comparison that retains or removes GPI from the 10-step teacher checkpoint.

| Teacher context | AIME24 | AIME25 | HMMT25 | Mean |
|---|---|---|---|---|
| With GPI | 53.3 | 42.2 | 31.1 | 42.2 |
| Without GPI | 52.8 | 40.3 | 29.7 | 40.9 |
| Difference (pp) | +0.6 | +2.0 | +1.4 | +1.3 |

Table 9 shows a longer early decline without GPI: validation avg@8 falls from checkpoint 10 to checkpoint 30 before recovering. With GPI, the validation avg@8 decreases only from checkpoint 10 to checkpoint 20, and after checkpoint 20 the GPI run is higher than the no-GPI run at every measured checkpoint. Together with the final evaluation, this trajectory supports retaining the original context during teacher evolution.

Table 9: Student validation trajectory after removing or retaining GPI from the 10-step checkpoint.

| Student checkpoint | No GPI avg@8 | With GPI avg@8 |
|---|---|---|
| 10 | 58.9 | 58.9 |
| 20 | 57.0 | 57.3 |
| 30 | 57.0 | 58.0 |
| 40 | 58.9 | 59.7 |
| 50 | 60.6 | 66.8 |
| 60 | 62.8 | 67.6 |
| 70 | 65.2 | 67.4 |
| 80 | 65.3 | 68.1 |
| 90 | 65.2 | 68.2 |

### C.1 Privileged Information Prompts

The procedural GPI consists of three task-level instructions covering problem interpretation, solution planning, and final verification. The teacher-only context is reproduced below.

> **R001 [Problem interpretation].** Before choosing a solution method, identify the exact target, all explicit constraints, and any domain restrictions stated or implied by the problem.
>
> **R002 [Solution planning].** Identify the main mathematical structure before performing detailed calculations. Prefer a coherent plan over unstructured trial and error.
>
> **R003 [Final verification].** Before finalizing, check the result against the original conditions, verify that transformations were valid, and check whether any cases or candidate solutions were omitted.

The off-topic privileged context consists of readable statements about coastal climate, soil texture, and autumn planting. This context is unrelated to mathematical reasoning and remains fixed throughout each corresponding control run.

> **R001 [Coastal climate].** Regions beside large bodies of water tend to have milder winters than inland regions at the same latitude, because water cools slowly.
>
> **R002 [Soil texture].** Loam combines sand, silt and clay in proportions that hold moisture well without leaving the ground badly waterlogged.
>
> **R003 [Autumn planting].** Bulbs are usually planted in autumn so that the necessary period of cold has passed before the ground warms and new growth begins again.

## D Optimizing Privileged Information from Student Trajectories

We also explored adapting the teacher's privileged information while keeping its parameters fixed. Terra analyzed sampled student trajectories, and its analysis informed revisions of the teacher-only context through a procedure inspired by Training-Free GRPO. Thus, teacher-side adaptation operated on the privileged context rather than the model parameters. We evaluated the resulting Qwen3-1.7B student using avg@12.

Table 10: Qwen3-1.7B student avg@12 accuracy (%) after Terra-guided privileged-information optimization with teacher parameters fixed.

| Method | AIME24 | AIME25 | HMMT25 | Mean |
|---|---|---|---|---|
| Terra-guided PI optimization | 54.2 | 37.2 | 25.0 | 38.8 |

Table 10 reports a macro avg@12 of 38.8% for context optimization, compared with 42.9% for OPED under fixed GPI (Table 2).