PROGRESS-AWARE ON-POLICY DISTILLATION FOR MULTIMODAL AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Standard on-policy distillation (OPD) applies a uniform objective across student-visited states. In long-horizon multimodal tasks, however, local teacher alignment can become decoupled from realized student progress. We identify *low-progress continuation*, where compact agents continue searching or invoking tools without meaningfully updating their answer state. We present **Δ-OPD**, a progress-aware distillation method that conditions teacher supervision on the student's realized progress. We quantify progress through the magnitude of the step-wise change in *answer readiness*, which measures the student's support for the reference answer at its current state. Δ-OPD retains reverse KL for progressing interactions and applies forward KL to low-progress interactions to strengthen supervision on teacher-supported alternatives. For verified-success trajectories, it masks low-progress continuation losses when readiness exceeds the trajectory-specific successful-answer reference. The method operates entirely on student-generated trajectories, requires no additional rollouts, and introduces no inference-time overhead. Experiments on OmniGAIA, VideoDR, and VisualSimpleQA show improved average accuracy for both Qwen3.5-4B and Qwen3.5-2B students. On OmniGAIA, Δ-OPD improves Qwen3.5-4B accuracy from **40.8%** under standard OPD to **45.4%**, while reducing the budget-exhaustion rate from **15.3%** to **9.8%**, a relative reduction of **35.9%**. Ablation studies show that selecting where to apply forward KL matters beyond its overall frequency, while readiness-based masking provides further gains. Code is provided in the supplementary material.

1 INTRODUCTION

Multimodal agents increasingly solve complex tasks by combining heterogeneous perception, long-context reasoning, and interaction with external tools. Recent omni-modal models and benchmarks extend this setting to tasks involving images, video, audio, web retrieval, and multi-turn tool use (Team, 2026b; Li et al., 2026b). Transferring these capabilities to compact models is particularly challenging because each interaction changes the evidence available to the agent and, consequently, the states it encounters next.

On-policy distillation (OPD) is well suited to this setting: the student generates its own trajectories while a stronger teacher provides dense token-level supervision at the states the student actually visits (Agarwal et al., 2024). Recent work has nevertheless revealed that long-horizon OPD can suffer from increasing interaction length, state-distribution drift, and uneven supervision across turns (Luo et al., 2026; Wang et al., 2026b; Li et al., 2026a; Zhong et al., 2026; Li et al., 2026c).

Standard OPD applies the same local distillation objective across student-visited states, effectively treating local imitation as equally useful throughout a trajectory. In long-horizon multimodal interaction, however, we find that local alignment with a stronger teacher can become decoupled from realized student progress. The consequence is a characteristic failure mode that we call *low-progress continuation*. Unlike length or repetition inflation, low-progress continuation concerns whether interaction changes the student's answer state: a long trajectory may remain productive, while a non-repetitive trajectory may already have stalled. These continuations need not be locally implausible. Another search or perception operation can remain well supported under the current history while the trajectory itself has stopped making progress. Repeatedly optimizing local agreement can

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.

therefore continue refining an interaction mode that is reasonable in isolation but no longer useful to the student.

A natural alternative is to modulate supervision by local teacher–student discrepancy. However, our behavioral analysis shows that local discrepancy is only weakly predictive of downstream interaction failure, whereas changes in the student's answer state provide a stronger signal (Section 2.2). This suggests that local alignment and realized interaction progress capture different properties of a visited state.

These observations lead to the central principle of our approach: *given the same teacher and the same student-visited state, supervision should depend on what the interaction actually accomplished for the student, rather than on local agreement with the teacher alone*. We characterize this effect through the student's *answer readiness*: how strongly it would support the reference answer if required to answer at the current state. The change in readiness across consecutive states measures the realized effect of an interaction. For successful trajectories, the readiness of the state from which the student produces a verified correct answer further provides a trajectory-specific reference for interpreting low-progress interaction.

We therefore introduce **Δ-OPD**, which makes realized student progress an explicit control signal for on-policy distillation. Interactions that continue to change the student's answer state retain reverse KL, refining behavior the student is already able to realize. Low-progress interactions use forward KL to expand supervision toward teacher-supported alternatives, unless the student's readiness already exceeds its trajectory-specific successful-answer reference, in which case continuation supervision is withheld. For trajectories without a verified correct completion, no readiness-based withholding is applied. Δ-OPD keeps training rollouts entirely student-generated, requires no additional teacher rollout, and introduces no inference-time component.

We evaluate Δ-OPD by distilling Qwen3.5 students on OmniGAIA. For Qwen3.5-4B, Δ-OPD improves task accuracy from **40.8%** under standard OPD to **45.4%**, while reducing the budget-exhaustion rate from **15.3%** to **9.8%**, a relative reduction of **35.9%**. Trajectory-level analyses further show that these gains are not explained by merely shortening trajectories: Δ-OPD spends less time in low-progress interaction regions while preserving productive long-horizon behavior. Across OmniGAIA and two out-of-distribution benchmarks, the gains also transfer to a smaller Qwen3.5-2B student, indicating that the proposed progress-conditioned distillation principle generalizes across student scales.

Our contributions are threefold:

- We identify *low-progress continuation* as a failure mode of long-horizon on-policy distillation, showing that local teacher alignment can become decoupled from realized student progress.
- We introduce Δ-OPD, which uses realized progress and a trajectory-specific readiness reference to adapt teacher supervision across interaction states.
- Experiments on OmniGAIA and two out-of-distribution benchmarks show that Δ-OPD improves task accuracy over standard OPD across student scales, while substantially reducing budget exhaustion and time spent in low-progress interaction.

2 Behavioral Motivation: From Premature Answering to Low-Progress Continuation

Before introducing our method, we examine how learning from a stronger teacher reshapes the interaction behavior of a compact multimodal agent. We train a Qwen3.5-4B student using Qwen3.8-27B as the teacher, either by supervised fine-tuning (SFT) on teacher trajectories or by standard on-policy distillation (OPD) on student-generated trajectories, and compare the resulting behavior with the teacher on OmniGAIA (Li et al., 2026b).

2.1 SFT and OPD Exhibit Opposite Failure Modes

SFT inherits a stopping policy that the student's capability cannot support. SFT learns interaction behavior from teacher-generated trajectories, including when the teacher chooses to stop. The student does not simply memorize a fixed interaction length: although the training trajectories

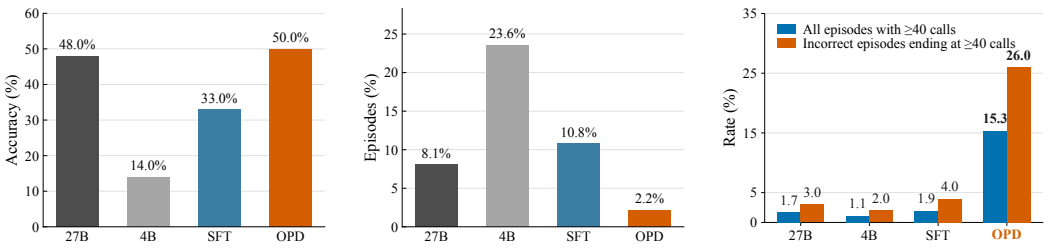

Figure 1: **SFT and OPD exhibit opposite failure modes.** We report zero-call accuracy, zero-call rate, and the fraction of abnormally long trajectories under SFT, standard OPD, and the teacher. SFT exhibits more premature answering, whereas OPD substantially reduces zero-call behavior but produces many more long trajectories.

contain an average of 3.6 tool calls on easier questions, the SFT student uses 7.8 calls on harder test questions, closely matching the teacher's 7.9. Its stopping behavior therefore adapts to task difficulty.

This behavioral alignment does not imply matched answer capability. Among questions answered without any tool call, the teacher is correct on 48%, whereas the SFT student is correct on only 33%. Moreover, among questions that SFT answers incorrectly without using a tool, an OPD student continues interacting and eventually solves 35%. SFT can therefore inherit the shape of the teacher's stopping behavior without acquiring the capability that makes the same stopping decisions reliable. We refer to this behavior as *premature answering*: the student commits to an answer even though further interaction can still recover a correct solution.

OPD corrects premature answering but overshoots into prolonged interaction. Standard OPD substantially corrects this failure. The zero-call rate drops from 10.8% under SFT to 2.2% under OPD, while zero-call accuracy rises from 33% to 50% (Figure 1). The behavioral shift, however, extends well beyond correcting premature answers. Average interaction length increases from 7.8 to 17.1 calls, and the fraction of trajectories exceeding 40 calls rises from 1.9% to 15.3%. Only 17% of web searches in these long trajectories are distinct, compared with 86% under SFT and the teacher. Within these long trajectories, the additional interaction is therefore dominated by redundant search rather than diverse evidence acquisition.

These statistics expose a behavioral tension rather than merely a preference for short or long trajectories. Standard OPD applies the same reverse-KL objective at every student-visited state. This objective can refine a productive interaction mode, but provides no mechanism for distinguishing it from a mode that the student keeps revisiting after its interaction has ceased to change the answer state. The issue is therefore not reverse KL in isolation, but its uniform application across states with different realized effects. This raises a more direct question: when the student continues to interact, is its answer state actually still changing?

2.2 Local Alignment Does Not Reveal Realized Progress

To understand whether continued interaction is actually productive, we inspect the student's *answer readiness*: how strongly the current state supports the reference answer if the student were required to answer immediately. We denote by Δ_t the readiness change induced by interaction t; its formal definition is deferred to Section 3.1.

Neither trajectory length nor surface repetition reliably identifies progress. With the threshold later used by our method ($|\Delta_t| \leq 0.05$), 39% of tool-calling turns exhibit little readiness change, yet 98% of them are not exact repetitions. Conversely, even after eight interaction turns, 45% of subsequent turns still produce changes above the threshold. Thus, long interaction can remain productive, while low-progress continuation often occurs without explicit repetition.

Local teacher–student discrepancy does not resolve this ambiguity either. Although it measures how far the student is from the teacher locally, it is much less informative about downstream trajectory behavior than readiness change (Appendix C). Moreover, trajectories observed at a low-progress state subsequently fail in 72% of cases and exhaust their interaction budget in 61%, compared with 58% and 42% at progressing states. These observations suggest that local teacher alignment and

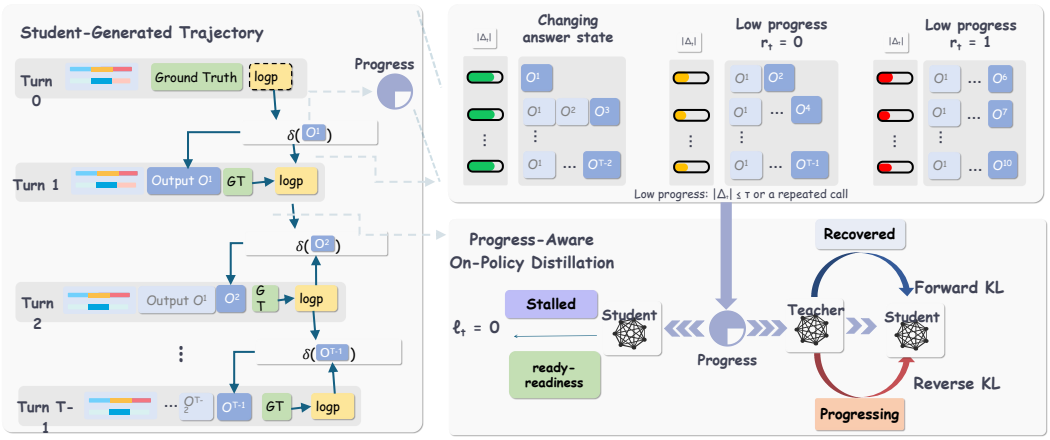

Figure 2: **Overview of Δ-OPD.** Realized progress and a trajectory-specific readiness reference determine whether each student-generated interaction turn receives reverse KL, forward KL, or zero loss. Terminal completions retain reverse KL, and losses are averaged within and across turns.

realized interaction progress capture distinct properties. They motivate conditioning distillation on the student's answer-readiness dynamics. We formalize these quantities and develop Δ-OPD in the next section.

3 Method

3.1 Answer Readiness and Realized Progress

Consider a student agent interacting with an environment over a multi-turn trajectory

$$\xi = (h_0, a_0, o_0, \ldots, h_T, a_T), \tag{1}$$

where h_t denotes the interaction history before assistant turn t, a_t denotes the student's action at that turn, and o_t is the resulting environment observation. We reserve a_T for the terminal completion. For each interaction turn $t < T$, the student produces a_t, the environment returns o_t, and the resulting interaction is appended to the history to form h_{t+1}. In on-policy distillation (OPD), trajectories are generated entirely by the student, while a stronger teacher provides token-level supervision at the student-visited states.

Section 2 shows that local teacher alignment does not reveal whether an interaction actually advances the student. We therefore characterize each visited state through the student's own answer state and use its dynamics to determine how the corresponding interaction is supervised.

Answer readiness. Let $y^* = (y_1^*, \ldots, y_m^*)$ be the reference answer of a training example. For every student-visited state h_t, we define its *answer readiness* as the average log-likelihood that the current student assigns to the reference answer if required to answer immediately:

$$V_t = \frac{1}{m} \sum_{i=1}^{m} \log \bar{\pi}_\theta \left(y_i^* \mid h_t \oplus \texttt{<answer>}, y_{<i}^* \right), \tag{2}$$

where m is the number of reference-answer tokens and $\bar{\pi}_\theta = \mathrm{sg}[\pi_\theta]$ denotes the current student policy evaluated with gradients stopped. The history h_t is the actual interaction context seen by the agent, including the multimodal task input, all preceding student turns, and the corresponding tool observations. For readiness scoring, we append the fixed `<answer>` prefix, condition on the reference prefix $y_{<i}^*$, and score only the reference-answer tokens.

Thus, V_t answers a counterfactual question: *if the student had to answer from the current state without further interaction, how strongly would its predictive distribution support the correct answer?* Equivalently, $-V_t$ is the average per-token surprisal of the reference answer under the current interaction context. Larger V_t therefore indicates stronger support for the correct answer.

Realized progress. We quantify the realized effect of interaction turn t through the change in answer readiness between its preceding and resulting states:

$$\Delta_t = V_{t+1} - V_t, \qquad t < T. \tag{3}$$

We use the magnitude of this change rather than its sign. New evidence need not increase answer readiness monotonically: an informative observation may invalidate the student's current hypothesis and temporarily reduce the likelihood of the reference answer before subsequent reasoning incorporates the new evidence. Both positive and negative changes can therefore indicate a substantive update to the student's answer state, whereas $|\Delta_t| \approx 0$ indicates that the interaction produced little change.

We refer to $|\Delta_t|$ as the measure of *realized progress*. Here, progress denotes substantive evolution of the student's answer state rather than monotonic improvement in reference-answer likelihood.

Low-progress interaction. For each interaction turn $t < T$, we define

$$z_t = \mathbb{I}\left[|\Delta_t| \le \tau\right] \vee \mathbb{I}\left[\mathrm{Repeat}_t\right], \tag{4}$$

where τ is a global progress threshold, set to 0.05 in the main experiments, and $\mathrm{Repeat}_t = 1$ when the tool call issued at turn t exactly matches an earlier call in the same episode, including both the tool name and its normalized arguments.

The two criteria capture complementary forms of low-progress interaction. The repeat indicator identifies explicitly redundant tool use, whereas $|\Delta_t|$ captures interactions that differ in surface form or action choice but nevertheless leave the student's answer state nearly unchanged. Examples include a reformulated search that retrieves essentially the same evidence or a visual interaction that revisits content the student has already inspected. Edge cases, including truncated turns and states for which readiness cannot be scored, are handled in Appendix B.

Because Δ_t compares the states immediately before and after turn t, the supervision assigned to that turn reflects its realized effect on the student's answer state. These quantities are computed after the student rollout is available and therefore require no additional rollout or routing decision during generation.

Readiness reference. Low progress alone does not identify a unique supervision regime. The behavioral analysis in Section 2.1 shows that teacher-shaped stopping behavior need not align with the student's own capability. We therefore anchor readiness to the student's own successful trajectory rather than to a teacher-derived global criterion.

For a trajectory whose final completion a_T is verified correct, let

$$V_{\mathrm{ans}} = V_T \tag{5}$$

denote the readiness at the state from which that answer is produced. For each interaction turn $t < T$, we define

$$r_t = \mathbb{I}\left[V_t > V_{\mathrm{ans}}\right]. \tag{6}$$

A turn with $r_t = 1$ starts from a state at which the student assigns higher likelihood to the reference answer than at the state from which it successfully answered. The resulting reference is trajectory-specific and introduces no additional globally tuned readiness threshold.

For trajectories without a verified correct completion, we set $r_t = 0$ for all interaction turns. Together, z_t and r_t distinguish interactions that continue to change the student's answer state from low-progress interactions, with the latter further separated according to whether they exceed the student's successful-answer readiness reference.

3.2 Δ-OPD: Progress-Aware On-Policy Distillation

For token position i of assistant turn t, let

$$q_{t,i} = \pi_\theta(\cdot \mid h_t, a_{t,<i}), \qquad p_{t,i} = \pi_{\mathrm{te}}(\cdot \mid h_t, a_{t,<i}) \tag{7}$$

denote the student and teacher next-token distributions under the same student-generated autoregressive prefix.

Δ-OPD uses realized progress and answer readiness to determine the supervision applied to each interaction turn:

$$\ell_{t,i} = \begin{cases} D_{\mathrm{KL}}\left(q_{t,i} \,\|\, p_{t,i}\right), & z_t = 0, \\ D_{\mathrm{KL}}\left(p_{t,i} \,\|\, q_{t,i}\right), & z_t = 1,\ r_t = 0, \\ 0, & z_t = 1,\ r_t = 1. \end{cases} \qquad t < T. \tag{8}$$

The decision is made once per turn and applied to all valid supervised tokens of that turn.

When $z_t = 0$, the interaction materially changes the student's answer state. We retain the reverse-KL objective of standard OPD, allowing the teacher to refine behavior that the student is already able to realize.

When $z_t = 1$ and $r_t = 0$, the student's current interaction has produced little realized progress but does not satisfy the readiness condition for withholding further interaction supervision. Forward KL transfers probability mass toward teacher-supported alternatives that the student currently underweights, while leaving rollout generation unchanged.

The distinction can be seen directly from the token-level gradients. Let q and p denote the student and teacher distributions at one token position, and let s_k be the student logit for token k:

$$\frac{\partial D_{\mathrm{KL}}(q\|p)}{\partial s_k} = q_k \left[\log \frac{q_k}{p_k} - D_{\mathrm{KL}}(q\|p)\right], \tag{9}$$

$$\frac{\partial D_{\mathrm{KL}}(p\|q)}{\partial s_k} = q_k - p_k. \tag{10}$$

For a teacher-preferred alternative with $q_k \ll p_k$, the reverse-KL correction is attenuated by the student's own probability q_k, whereas the forward-KL gradient remains approximately $-p_k$ even when the student assigns little probability to that alternative. Forward KL therefore preserves corrective pressure on precisely the teacher-supported alternatives that the student's current interaction mode underweights.

When $z_t = 1$ and $r_t = 1$, the interaction produces little realized progress from a state whose readiness already exceeds the student's successful-answer reference. We therefore assign that turn zero distillation loss rather than further reinforcing its continuation. We do not replace this signal with an explicit stopping target: readiness measures support for the reference answer but does not certify what the student would actually generate if it stopped.

Δ-OPD does not insert teacher-generated actions into the training trajectory or require a separate teacher continuation. It changes only how the teacher distribution supervises student-generated turns after the rollout has been collected.

3.3 Training Objective

Δ-OPD applies the progress-conditioned distillation objective at the interaction-turn level and uses turn-normalized loss aggregation. Let $\mathcal{M}_t$ denote the set of valid assistant-token positions in turn t under the standard training mask. For each turn, we first average the token-level distillation losses:

$$\mathcal{L}_t = \frac{1}{|\mathcal{M}_t|} \sum_{i \in \mathcal{M}_t} \ell_{t,i}. \tag{11}$$

For interaction turns $t < T$, the token-level loss $\ell_{t,i}$ is determined by the routing rule in Eq. (8).

For the terminal completion, we retain the standard reverse-KL objective:

$$\ell_{T,i} = D_{\mathrm{KL}}\left(q_{T,i} \,\|\, p_{T,i}\right). \tag{12}$$

The trajectory-level objective then assigns equal weight to each assistant turn:

$$\mathcal{L}_{\Delta\text{-OPD}} = \frac{1}{T+1} \sum_{t=0}^{T} \mathcal{L}_t. \tag{13}$$

This turn-normalized reduction prevents long interaction turns from dominating the optimization merely because they contain more tokens, while preserving the progress-conditioned supervision assigned to each turn.

Model / Method	Category-Wise Breakdown									Overall
	Geo.	Tech.	Hist.	Fin.	Sport	Art	Movie	Sci.	Food	
Proprietary Models										
Gemini-3-Pro	65.2	59.2	62.1	72.0	78.4	52.8	48.5	42.3	88.9	62.5
Open-source General Models										
Qwen2.5-Omni-3B	0.0	2.0	4.5	0.0	0.0	0.0	0.0	3.9	0.0	1.4
Qwen2.5-Omni-7B	1.5	4.1	7.5	4.0	0.0	2.8	0.0	7.7	5.6	3.6
Qwen3-Omni-30B-A3B	8.7	14.3	11.9	28.0	10.8	13.9	9.1	15.4	22.2	13.3
Open-source Agentic Models										
OmniAtlas-Qwen2.5-7B	8.7	18.4	16.4	4.0	16.2	22.2	3.0	7.7	22.2	13.3
OmniAtlas-Qwen3-30B-A3B	10.1	30.6	29.9	32.0	18.9	16.7	12.1	11.5	27.8	20.8
Orchestra-o1-8B	21.7	32.7	37.9	12.0	29.7	16.7	45.5	38.5	38.9	30.0
Qwen3.5-4B Training Methods										
Qwen3.5-4B	26.1	38.8	35.9	44.0	32.4	38.0	23.2	29.5	29.6	33.0
Qwen3.5-4B + SFT	29.7	41.8	48.5	52.0	37.8	30.6	**42.4**	28.8	50.0	39.6
Qwen3.5-4B + OPD	31.9	41.8	51.5	44.0	**41.9**	36.1	36.4	34.6	**52.8**	40.8
Qwen3.5-4B + Δ-OPD (Ours)	**36.2**	**52.0**	**53.8**	**56.0**	39.2	**44.4**	36.4	**46.2**	44.4	**45.4**

Table 1: **Category-wise performance on OmniGAIA.** Mean accuracy (%) over four runs across nine domains. Prior-model results are reproduced from previously reported evaluations; bold indicates the best Qwen3.5-4B training result in each column.

Both KL directions are computed from the full next-token distributions rather than from the sampled token alone. Answer readiness and the resulting supervision decisions are used only during training. At inference time, neither the reference answer nor readiness computation is required, and the trained student is deployed in the same manner as a standard OPD-trained policy. Implementation details, including masking and turn-level aggregation, are provided in Appendix B. Figure 2 summarizes how Δ-OPD uses realized progress and a trajectory-specific readiness reference to determine turn-level distillation supervision.

4 Experiments

4.1 Experimental Setup

Data and Benchmarks. We train on 2,156 tasks released with OmniGAIA (Li et al., 2026b). For OPD-based methods, students generate trajectories on-policy from the original queries and multimodal inputs, while SFT uses correctness-filtered trajectories generated by Qwen3.8-27B. We evaluate on OmniGAIA and two out-of-distribution benchmarks, VideoDR (Liu et al., 2026) and VisualSimpleQA (Wang et al., 2025), covering omni-modal, video, and image-based agent tasks.

Models and Baselines. Our primary setting uses Qwen3.8-27B (Qwen Team, 2026) as the teacher and Qwen3.5-4B (Team, 2026a) as the student. We additionally evaluate Qwen3.5-2B under the same teacher to examine generalization across student scales. Both students use released post-trained checkpoints with native tool-use support; thinking mode is disabled throughout training and evaluation. We compare Δ-OPD with the base student, SFT, standard OPD (Agarwal et al., 2024), and recent multi-turn OPD methods, including TCOD (Wang et al., 2026b), SOD (Zhong et al., 2026), and TurnOPD (Zhou et al., 2026).

Evaluation Protocol. All methods use matched training and interaction settings whenever applicable and are evaluated with identical prompts, decoding configurations, tool access, and interaction budgets. We report mean accuracy over four independent runs for each benchmark, together with the unweighted average across the three benchmarks. Full experimental and evaluation details are provided in Appendix A.

Table 2: **Cross-benchmark comparison across student scales.** Mean task accuracy (%) over four runs. **Overall** is the full-benchmark score and **Avg.** is the unweighted mean of the three Overall scores. For VisualSimpleQA, **Hard** is the official 129-question subset and **Normal** contains the remaining 371 questions. Bold indicates the best training result within each student setting.

Model / Method	OmniGAIA				VideoDR				VisualSimpleQA			Avg.
	Easy	Medium	Hard	Overall	Easy	Medium	Hard	Overall	Normal	Hard	Overall	
Teacher Reference												
Qwen3.8-27B	62.7	54.4	41.7	54.4	83.3	50.9	43.8	59.0	78.1	41.3	68.6	60.7
Student: Qwen3.5-4B												
Qwen3.5-4B	45.6	29.6	20.1	33.0	50.0	34.3	15.6	33.3	65.4	40.3	58.9	41.7
OPD	55.7	36.2	26.9	40.8	73.4	31.9	30.5	44.8	73.0	**48.1**	66.6	50.7
TCOD	58.6	40.3	25.6	43.3	70.3	**43.8**	26.6	46.8	75.5	42.6	67.0	52.4
SOD	56.6	38.1	**31.4**	42.9	67.2	37.5	32.8	45.5	77.4	40.3	67.8	52.1
TurnOPD	56.1	39.1	30.8	43.1	70.3	26.4	33.6	42.8	74.9	43.4	66.8	50.9
Δ-OPD (Ours)	**59.4**	**43.1**	28.2	**45.4**	**75.8**	42.4	**38.3**	**51.8**	**77.6**	45.0	**69.2**	**55.5**
Student: Qwen3.5-2B												
Qwen3.5-2B	19.7	14.7	9.0	15.1	43.8	26.4	21.9	30.5	52.2	22.9	44.6	30.1
OPD	33.6	18.4	9.6	21.7	54.7	27.1	**23.4**	34.8	64.4	**34.1**	56.6	37.7
TCOD	32.8	18.1	10.3	21.4	53.9	20.8	7.8	27.3	62.8	31.8	54.8	34.5
SOD	**36.1**	20.0	9.0	23.1	57.8	**34.7**	14.8	**35.8**	59.0	32.6	52.2	37.0
TurnOPD	32.4	15.9	9.0	20.0	**64.1**	23.6	16.4	34.3	65.0	30.2	56.0	36.8
Δ-OPD (Ours)	34.4	**23.1**	**14.1**	**25.0**	57.8	31.9	10.2	33.3	**66.8**	**34.1**	**58.4**	**38.9**

4.2 Main Results

Table 1 reports category-wise performance on OmniGAIA. Δ-OPD reaches **45.4%** overall, improving standard OPD by 4.6 points ($40.8\% \rightarrow 45.4\%$), SFT by 5.8 points, and the original student by 12.4 points. The gain is broad: Δ-OPD obtains the best result among Qwen3.5-4B training methods in six of nine domains, with particularly large improvements over OPD in Finance ($+12.0$), Science ($+11.6$), and Technology ($+10.2$).

The resulting 4B student also exceeds previously reported open-source agentic models, reaching 45.4% versus 30.0% for Orchestra-o1-8B and 20.8% for OmniAtlas-Qwen3-30B-A3B. Since these results follow their originally reported evaluation settings, we treat them as reference points rather than controlled comparisons.

Table 2 shows that the gains extend beyond OmniGAIA. For the 4B student, Δ-OPD achieves the best Overall score on all three benchmarks, improving OPD by 4.6, 7.0, and 2.6 points on OmniGAIA, VideoDR, and VisualSimpleQA, respectively. Its cross-benchmark average reaches **55.5%**, exceeding OPD by 4.8 points and the strongest competing distillation baseline by 3.1 points.

The advantage largely persists at 2B. Δ-OPD improves OPD from 21.7% to 25.0% on OmniGAIA and from 56.6% to 58.4% on VisualSimpleQA, yielding the highest cross-benchmark average of **38.9%**. VideoDR is the exception, where SOD performs best (35.8% versus 33.3%). Overall, the results show that progress-conditioned distillation transfers across benchmark distributions and student scales, although the advantage is not uniform across every individual setting.

4.3 Ablation Study

Table 3 isolates the contribution of each component. Removing FKL and simply dropping the corresponding low-progress supervision reduces the average from 55.5% to 52.8%, showing that low-progress states benefit from teacher-covering supervision rather than being ignored. More importantly, randomly assigning FKL while preserving its original frequency further lowers the average to 52.1%. Thus, the gain does not arise from merely mixing forward and reverse KL. Where FKL is applied, as determined by realized progress, is critical. Readiness-based masking provides an additional 1.3-point gain, supporting the removal of redundant continuation supervision after the trajectory-specific readiness reference has been reached. Turn-level normalization contributes a smaller but consistent 0.8-point gain, indicating that balancing supervision across turns further stabilizes the objective. Together, the ablations show that the main improvement comes from progress-conditioned

Table 3: **Ablation study of Δ-OPD.** All variants use Qwen3.8-27B as the teacher and Qwen3.5-4B as the student. Mean task accuracy (%) is reported over four runs; **Avg.** is the unweighted average across the three benchmarks. Random FKL preserves the original FKL proportion within each trajectory while randomizing the turns to which it is assigned.

Method	OmniGAIA	VideoDR	VisualSimpleQA	Avg.
Δ-OPD (Full)	**45.4**	**51.8**	**69.2**	**55.5**
w/o FKL (replaced with zero loss)	43.0	48.2	67.2	52.8
w/o Progress-Based Routing (Random FKL)	42.3	47.3	66.7	52.1
w/o Readiness-Based Masking	44.1	50.1	68.3	54.2
w/o Turn-Level Normalization	44.7	50.9	68.6	54.7

KL routing, with readiness-based masking and turn-level normalization providing complementary gains.

5 Related Work

Multimodal Agent. LLM-driven autonomous agents tackle Long-horizon tasks by observing, reasoning and acting via external tools. Wang et al. (2026a) proposed IGPO which derives intrinsic rewards directly from the model's own belief updates. These intrinsic turn-level rewards are combined with outcome-level supervision to form dense reward signals.

Knowledge Distillation. Recent studies identify severe state drift as a primary bottleneck in vanilla OPD (Luo et al., 2026; Wang et al., 2026b; Li et al., 2026a; Zhong et al., 2026). Early student mistakes compound rapidly, pushing subsequent trajectories outside the teacher's reliable support. This drift often manifests as repetition loops (Luo et al., 2026), exhausted search budgets (Li et al., 2026a), and superlinear divergence accumulation Wang et al. (2026b); Zhong et al. (2026). To mitigate this, existing works restrict long-horizon exploration via temporal curricula (Wang et al., 2026b), probabilistically inject teacher actions to recover familiar states (Li et al., 2026a), or downweight later turns based on teacher-student divergence (Zhong et al., 2026). Instead of relying on rigid curricula or external heuristics, we dynamically resolve stagnation by conditioning supervision strictly on the student's own progress.

6 Conclusion

In this work, we study how on-policy distillation can better support long-horizon multimodal agents. Our analysis highlights a limitation of uniform teacher supervision: closer teacher alignment does not necessarily translate into productive interaction. We propose Δ-OPD, which uses realized changes in answer readiness and a trajectory-specific reference to assign reverse KL, forward KL, or zero loss to individual interaction turns. Together with turn-level normalization, this design adapts supervision to the student's evolving answer state. Experiments on OmniGAIA and two out-of-distribution benchmarks show improved average accuracy across two student scales. On OmniGAIA, Δ-OPD improves Qwen3.5-4B from 40.8% to 45.4% accuracy while reducing budget exhaustion by 35.9% relative to standard OPD. Ablations support the importance of progress-conditioned KL routing, with additional gains from readiness-based masking and turn-level normalization. These findings highlight the value of using interaction progress to guide agent distillation. Future work will explore progress estimation without reference answers and evaluate transfer across model families and broader agent environments.

AI use statement

Generative AI tools were used solely to improve the clarity, readability, and organization of the manuscript. All scientific content, experimental design, implementation, analysis, and conclusions were produced and verified by the authors.

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.

Gengsheng Li, Mao Zheng, Mingyang Song, Ruiqi Liu, Tianyu Yang, Jie Sun, Qiyong Zhong, Haiyun Guo, Junfeng Fang, Dan Zhang, et al. On-policy distillation with curriculum turn-level guidance for multi-turn agents. *arXiv preprint arXiv:2606.15912*, 2026a.

Xiaoxi Li, Wenxiang Jiao, Jiarui Jin, Haoxuan Li, Hao Wang, Shijian Wang, Guanting Dong, Jiajie Jin, Yinuo Wang, Yuan Lu, et al. Omnigaia: Towards native omni-modal ai agents. *arXiv preprint arXiv:2602.22897*, 2026b.

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*, 2026c.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

Chengwen Liu, Xiaomin Yu, Zhuoyue Chang, Zhe Huang, Shuo Zhang, Heng Lian, Jisheng Dang, Rui Xu, Sen Hu, Jianheng Hou, et al. Watching, reasoning, and searching: A video deep research benchmark on open web for agentic video reasoning. *arXiv preprint arXiv:2601.06943*, 2026.

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.

Qwen Team. Qwen3.8-max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3. 5-omni technical report. *arXiv preprint arXiv:2604.15804*, 2026b.

Guoqing Wang, Sunhao Dai, Guangze Ye, Zeyu Gan, Wei Yao, Yong Deng, Xiaofeng Wu, et al. Information gain-based policy optimization: A simple and effective approach for multi-turn search agents. In *International Conference on Learning Representations*, volume 2026, pp. 104502–104529, 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.

Yanling Wang, Yihan Zhao, Xiaodong Chen, Shasha Guo, Lixin Liu, Haoyang Li, Yong Xiao, Jing Zhang, Qi Li, and Ke Xu. Visualsimpleqa: A benchmark for decoupled evaluation of large vision-language models in fact-seeking question answering. *arXiv preprint arXiv:2503.06492*, 2025.

Qiyong Zhong, Mao Zheng, Mingyang Song, Xin Lin, Jie Sun, Houcheng Jiang, Xiang Wang, and Junfeng Fang. Sod: Step-wise on-policy distillation for small language model agents. *arXiv preprint arXiv:2605.07725*, 2026.

Yuhang Zhou, Kai Zheng, Haoling Li, Dengyun Peng, Can Xu, and Jingjing Chen. Turnopd: Making on-policy distillation turn-aware for efficient long-horizon agent training. *arXiv preprint arXiv:2607.05804*, 2026.

A Additional Experimental Details

A.1 Training Data and Benchmarks

Training Data. We train on the 2,156 tasks released with OmniGAIA (Li et al., 2026b). For all OPD-based methods, the student starts from the original task query and associated multimodal inputs and generates its interaction trajectories on-policy.

For the SFT baseline, we independently generate trajectories using Qwen3.8-27B on the same training tasks and retain only those whose final answers are verified as correct. The resulting successful trajectories are used to fine-tune the Qwen3.5-4B student. SFT and OPD-based methods therefore share the same underlying training tasks, while differing in both trajectory collection and training objective.

OmniGAIA. OmniGAIA (Li et al., 2026b) is our primary evaluation benchmark. It contains 360 omni-modal agent tasks requiring long-horizon reasoning over images, audio, videos, and externally retrieved information.

VideoDR. VideoDR (Liu et al., 2026) evaluates video-conditioned deep research, requiring agents to identify informative visual evidence across video frames and combine it with information retrieved from the open web. We use the 100-question evaluation set and perform no benchmark-specific training or adaptation.

VisualSimpleQA. VisualSimpleQA (Wang et al., 2025) evaluates fact-seeking question answering grounded in visual inputs. We use the full 500-question set and additionally evaluate its 129-question hard subset.

A.2 Models

Our primary setting uses Qwen3.8-27B (Qwen Team, 2026) as the teacher and Qwen3.5-4B (Team, 2026a) as the student. We additionally evaluate Qwen3.8-27B$\rightarrow$Qwen3.5-2B to examine whether the behavior of Δ-OPD persists when the student capacity is reduced while keeping the teacher fixed. All Qwen3.5 students are used with thinking mode disabled during both training and evaluation.

A.3 Baseline Details

Unless otherwise specified, all methods start from the same student checkpoint, use the same training tasks, and interact with the same environment.

Base. The original Qwen3.5 student without additional task-specific post-training.

SFT. We perform supervised fine-tuning on correctness-filtered trajectories generated by Qwen3.8-27B. Unlike OPD-based methods, SFT trains on a fixed set of successful teacher-generated trajectories rather than trajectories induced by the current student policy.

OPD. We implement standard on-policy distillation following Agarwal et al. (2024). The student first generates its own trajectory, after which the teacher provides token-level supervision at the states visited by the student. The same reverse-KL objective is applied throughout the trajectory.

TCOD. TCOD (Wang et al., 2026b) introduces a temporal curriculum for multi-turn on-policy distillation by progressively adjusting the rollout horizon during training. Since our setting uses exclusively student-generated rollouts and does not incorporate teacher-generated trajectories, we compare against its forward-to-backward (F2B) variant under the same student–teacher and interaction setting as our method.

SOD. SOD (Zhong et al., 2026) performs step-wise on-policy distillation and modulates teacher supervision according to the student–teacher discrepancy at individual interaction steps. We implement

Table 4: Default training hyperparameters for online distillation. The progress threshold applies only to Δ-OPD.

Parameter	Value
Training Steps	100
Learning Rate	1×10^{-6}
Learning Rate Schedule	Constant
Training Batch Size	64
Rollouts per Prompt (n)	1
Rollout Temperature	1.0
Distillation Top-k	64
Progress Threshold (τ)	0.05

it under the same student-generated rollout setting as standard OPD and set its step-wise grouping parameter to $n = 4$.

TurnOPD. TurnOPD (Zhou et al., 2026) makes on-policy distillation turn-aware through adaptive rollout-depth control and turn-level loss normalization. As no official implementation is publicly available, we reimplement TurnOPD strictly following the method and training procedure described in the original paper, within our common training framework.

Δ-OPD. Our method follows the same student-generated rollout paradigm as standard OPD, but conditions the distillation objective on the realized answer-readiness dynamics of individual interactions. Interactions that induce a meaningful change in the student's answer state use reverse KL. Low-progress interactions use forward KL unless their readiness exceeds the trajectory-specific reference point, in which case their continuation distillation loss is removed. The terminal completion always retains the standard reverse-KL objective. Detailed routing and implementation are provided in Appendix B.

A.4 Training Details

We implement all methods using the `verl` framework (Sheng et al., 2025). Within each teacher–student configuration, online distillation methods share the same student initialization, training task pool, interaction environment, and available tools. Table 4 summarizes the default training configuration; method-specific settings are described in Appendix A.3. The distillation top-k denotes the number of vocabulary tokens retained when computing the KL loss. No hyperparameters are tuned on the out-of-distribution benchmarks.

A.5 Evaluation Details

For each trained model, we conduct four independent evaluation runs and sample one interaction trajectory per query in each run. We report the mean accuracy across the four runs. The overall "Avg." score is the unweighted arithmetic mean of the accuracies on OmniGAIA, VideoDR, and VisualSimpleQA.

For OmniGAIA, we follow its official Pass@1 evaluation protocol (Li et al., 2026b). Exact answer matches are directly accepted, while non-exact predictions are judged for semantic equivalence using DeepSeek-V3.2 (Liu et al., 2025).

Within each benchmark, all compared methods use identical prompts, decoding settings, tool access, and maximum interaction budgets. For analyses of interaction efficiency, we additionally report the average number of tool calls and the budget-exhaustion rate, defined as the fraction of episodes that exhaust the maximum interaction budget before producing a final answer.

B Implementation Details

B.1 Training Procedure

Algorithm 1 summarizes the training procedure of Δ-OPD. Each training iteration first collects trajectories generated entirely by the student. After the rollout, we evaluate answer readiness at the visited states, compute the realized progress of each interaction, and assign its distillation objective according to Eqs. (4)–(8). Token-level distillation losses are first averaged within each assistant turn, and the resulting turn-level losses are then averaged across the trajectory. No teacher-generated action is inserted into the training rollout.

Algorithm 1 Training procedure of Δ-OPD

Input: student π_θ, teacher π_{te}, training batch with reference answers, progress threshold τ

for each task (x, y^*) in the batch **do**
  Roll out the student trajectory $\xi = (h_0, a_0, o_0, \ldots, h_T, a_T)$
  Compute answer-readiness values $V_0, \ldots, V_T$ using Eq. (2)
  if a_T is verified correct **then**
    $V_{\text{ans}} \leftarrow V_T$
  end if
  for $t = 0, \ldots, T-1$ **do**
    $\Delta_t \leftarrow V_{t+1} - V_t$
    $z_t \leftarrow \mathbb{I}[|\Delta_t| \le \tau] \vee \mathbb{I}[\text{Repeat}_t]$
    if a_T is verified correct **then**
      $r_t \leftarrow \mathbb{I}[V_t > V_{\text{ans}}]$
    else
      $r_t \leftarrow 0$
    end if
    if $z_t = 0$ **then**
      Assign reverse-KL loss to interaction turn t
    else if $r_t = 0$ **then**
      Assign forward-KL loss to interaction turn t
    else
      Set the distillation loss of interaction turn t to zero
    end if
  end for
  Assign standard reverse-KL loss to terminal completion a_T
  Average token losses within each turn
  Average turn-level losses over the trajectory
end for
Average trajectory losses over the batch and update θ

B.2 Answer-Readiness Computation

For each visited state h_t, answer readiness is evaluated using the same student parameters that generated the trajectory and before the optimizer update for the current batch. The computation is performed with gradients stopped.

The state h_t is the actual interaction context presented to the model at turn t, including the multimodal task input, all preceding student turns, and the corresponding tool observations. For readiness scoring, we append the fixed `<answer>` prefix, condition on the reference prefix $y^*_{<i}$, and evaluate the reference-answer tokens autoregressively:

$$V_t = \frac{1}{m} \sum_{i=1}^{m} \log \bar{\pi}_\theta \left(y_i^* \mid h_t \oplus \texttt{<answer>}, y^*_{<i} \right), \tag{14}$$

where $\bar{\pi}_\theta = \text{sg}[\pi_\theta]$. Only the m reference-answer tokens contribute to the readiness score, and the same answer format is used at every visited state of an example.

This construction measures how strongly the student would support the reference answer if required to answer directly from the current interaction state, without additional reasoning or tool use. Readiness is used only as a training-time diagnostic and does not modify the student rollout.

B.3 Interaction-Level Decisions

For each interaction turn $t < T$, realized progress is computed from the change in answer readiness between the states immediately before and after the interaction:

$$\Delta_t = V_{t+1} - V_t. \tag{15}$$

The low-progress indicator combines the magnitude of this change with an exact-repeat criterion:

$$z_t = \mathbb{I}[|\Delta_t| \le \tau] \vee \mathbb{I}[\mathrm{Repeat}_t]. \tag{16}$$

Repeat_t is activated when the tool call issued at turn t exactly matches an earlier call in the same episode, including both the tool name and its normalized arguments. Thus, the repeat criterion captures explicitly redundant tool use, while the readiness-change criterion can identify interactions that differ in surface form but nevertheless leave the student's answer state nearly unchanged.

For trajectories with a verified correct terminal completion, we set

$$V_{\mathrm{ans}} = V_T \tag{17}$$

and compute the trajectory-specific readiness indicator

$$r_t = \mathbb{I}[V_t > V_{\mathrm{ans}}], \qquad t < T. \tag{18}$$

For trajectories without a verified correct completion, we set $r_t = 0$ for all interaction turns, as in the main method.

Both z_t and r_t are computed only after the complete student trajectory has been collected. They therefore affect the distillation objective assigned to an observed student-generated turn, rather than the rollout policy used to generate that turn.

B.4 Token-Level Distillation and Turn Normalization

For each supervised token position i of interaction turn t, the student and teacher are evaluated under the same student-generated autoregressive prefix:

$$q_{t,i} = \pi_\theta(\cdot \mid h_t, a_{t,<i}), \qquad p_{t,i} = \pi_{\mathrm{te}}(\cdot \mid h_t, a_{t,<i}). \tag{19}$$

For interaction turns $t < T$, the token-level objective follows Eq. (8): reverse KL is used when $z_t = 0$, forward KL is used when $z_t = 1$ and $r_t = 0$, and the distillation loss is set to zero when $z_t = 1$ and $r_t = 1$. All supervised tokens within the same interaction turn share the same interaction-level decision.

The terminal completion a_T does not receive an interaction-level progress decision and retains the standard reverse-KL objective:

$$\ell_{T,i} = D_{\mathrm{KL}}\left(q_{T,i} \,\|\, p_{T,i}\right). \tag{20}$$

To prevent turns with more generated tokens from receiving disproportionately large optimization weight, we use turn-normalized loss aggregation. Let $\mathcal{M}_t$ denote the set of valid assistant-token positions in turn t under the training mask. We first average the token-level losses within each turn:

$$\mathcal{L}_t = \frac{1}{|\mathcal{M}_t|} \sum_{i \in \mathcal{M}_t} \ell_{t,i}. \tag{21}$$

The trajectory loss is then computed by assigning equal weight to all assistant turns, including the terminal completion:

$$\mathcal{L}_{\mathrm{traj}} = \frac{1}{T+1} \sum_{t=0}^{T} \mathcal{L}_t. \tag{22}$$

Turns routed to zero continuation supervision have $\mathcal{L}_t = 0$ and remain part of the turn-level reduction. Finally, trajectory losses are averaged across examples in the training batch.

This reduction separates the choice of distillation direction from the amount of text generated at a turn: the routing rule determines *how* a state is supervised, while turn normalization prevents long responses from dominating the update solely because they contain more tokens.

Both KL directions are computed from the full next-token distributions rather than from the sampled token alone. Answer-readiness computation, progress routing, and the teacher distribution are required only during training; none introduces an inference-time component.

B.5 Boundary Cases

Incomplete interactions, truncated trajectories, and states for which answer readiness cannot be evaluated are handled conservatively so that they do not introduce spurious progress signals or readiness decisions. Exact handling rules for these implementation-specific cases are provided with the released code.

C Additional Analysis of Realized Progress

We provide additional evidence for the central observation in Section 2.2: local teacher–student alignment does not reveal whether the student's interaction is producing realized progress. Our analysis covers $28{,}952$ tool-calling turns from $6{,}400$ on-policy rollouts of the Qwen3.5-4B student over $2{,}139$ training questions, collected during on-policy distillation with a turn budget of 16. Every visited state is annotated with its readiness change Δ_t, the local teacher–student discrepancy seen by OPD, and the outcome of the remaining trajectory.

Length and repetition do not identify realized progress. Using the same threshold adopted by our method, $|\Delta_t| \leq 0.05$, 39% of tool-calling turns fall into the low-progress regime. However, 98% of these turns do not exactly repeat any earlier tool call, while exact repetitions account for only 1.3% of all tool-calling turns. Thus, explicit repetition captures only a small fraction of interactions that leave the student's answer state nearly unchanged.

Trajectory length is similarly insufficient. Even after 8 interaction turns, 45% of subsequent turns still exhibit readiness changes above the threshold. Long trajectories therefore contain both productive interactions and low-progress continuation, and interaction count alone cannot determine whether the student's answer state is still changing.

Local discrepancy does not reveal realized progress. A natural alternative is the local teacher–student discrepancy already computed during OPD. We measure discrepancy as the per-turn mean of the token-level reverse KL between student and teacher, evaluated under the same student-generated autoregressive prefixes and the teacher top-64 support used by the OPD objective.

Local discrepancy and realized progress capture fundamentally different properties. A student can disagree strongly with the teacher while still acquiring useful evidence, or remain locally aligned while continued interaction produces little change in its answer state. We test this distinction using downstream outcomes that do not depend on the low-progress criterion itself. At every visited state, we evaluate how well local discrepancy and $|\Delta_t|$ predict whether the remaining trajectory eventually fails, exhausts its interaction budget, or enters exact repetition.

As shown in Table 5, readiness dynamics are consistently more informative than local teacher–student discrepancy. For eventual task failure, $|\Delta_t|$ reaches an AUROC of 0.65, compared with only 0.47 for local discrepancy. For budget exhaustion, the corresponding values are 0.64 and 0.46. For future exact repetition, readiness change reaches 0.56, while local discrepancy reaches 0.45. The below-chance performance of local discrepancy further shows that large local student–teacher disagreement is not evidence of unproductive interaction; if anything, it is slightly associated with states in which the student's behavior is still changing.

Low-progress states predict undesirable continuation. The same separation is visible after discretizing interactions with the readiness-change threshold. A trajectory observed at a low-progress

Table 5: **Local discrepancy does not reveal realized progress.** AUROC of per-state signals for downstream outcomes measured after the same visited state ($28{,}952$ tool-calling turns from $6{,}400$ rollouts). For local discrepancy, larger disagreement is scored toward failure; for $|\Delta_t|$, smaller readiness change is scored toward failure.

Signal	Task Failure	Budget Exhaustion	Future Repetition
Local teacher–student discrepancy	0.47	0.46	0.45
Readiness change $\lvert\Delta_t\rvert$	**0.65**	**0.64**	**0.56**

state subsequently fails in 72% of cases and exhausts its interaction budget in 61%. For trajectories observed at a progressing state, the corresponding rates are only 58% and 42%. Thus, low progress is not merely a geometric property of the student's distribution: it is strongly associated with the downstream failure modes that characterize inefficient long-horizon interaction.

Taken together, these results expose a missing variable in standard OPD. Local teacher–student discrepancy tells us how the student differs from the teacher at the current state, but not whether the student's own interaction is still producing useful change. Realized readiness dynamics reveal this distinction, motivating the progress-conditioned supervision introduced by Δ-OPD.