WHEN AND WHERE SHOULD TEACHERS STOP TEACHING? LOCAL DISCRIMINABILITY COLLAPSE IN STRONG-TO-WEAK ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In strong-to-weak on-policy distillation (OPD), where a high-capacity teacher supervises a substantially smaller student on student-generated rollouts with dense token-level feedback, conventional practice assumes that supervising every generated token is uniformly beneficial whenever teacher guidance is accessible. In this paper, we uncover a prevalent and trajectory-specific failure mode that overturns this premise: *local discriminability collapse*. Specifically, while the global teacher-student advantage (the mean log-probability gap on sampled tokens) persists late into a response, the teacher's local discriminative contrast across the student's reachable alternative actions sharply decays, forcing standard full-trajectory OPD to optimize over a low-contrast, nearly uniform suffix signal. To resolve this mismatch, we introduce **Local-Discriminability-Aware On-Policy Distillation (LDA-OPD)**, a framework that selectively restricts dense teacher supervision to trajectory regions where guidance remains locally discriminative. Concretely, LDA-OPD applies three steps to each trajectory: it evaluates the teacher's log-probability margin between the top-two competitors within the student's high-probability candidate set, aggregates these margins over sentence-level semantic segments, and identifies the downward transition using a lightweight change-point test based on the profiled Bayesian Information Criterion (BIC) that is sweep-free over cutoff lengths. Past this boundary, we release dense supervision by zeroing the token-level loss on the suffix, and preserve per-sample loss mass by rescaling retained-prefix advantages so that the total loss magnitude remains unchanged. This adaptively truncates uninformative suffixes in a single training pass without requiring hyperparameter sweeps over cutoff lengths, while leaving student inference-time generation unconstrained. Extensive evaluations across multiple student scales (1.7B, 4B, 8B), model families (Qwen3, Gemma3), and diverse task types (multi-step mathematical reasoning, programmatic code generation, and out-of-domain scientific reasoning and instruction following) show that LDA-OPD consistently outperforms standard full-trajectory OPD and matches or surpasses multi-run fixed-prefix oracles in a single training run. Furthermore, matched-framework signal ablations and multi-seed evaluations provide strong empirical and counterfactual evidence that these gains are driven by local action contrast rather than generic entropy or position heuristics, establishing that effective strong-to-weak distillation requires evaluating not merely the presence of teacher guidance, but its local discriminative sharpness.

1 INTRODUCTION

When a strong teacher is available, distillation is an effective means of improving student performance. In conventional off-policy distillation, the student learns next-token prediction on teacher-generated prefixes, yet at test time it can condition only on prefixes it has produced itself; as these **prefixes drift**, the student may fail to sustain the teacher's output pattern (Liu et al., 2025b; Yan et al., 2026). Relative to off-policy distillation, on-policy distillation (OPD) (Agarwal et al., 2024) supervises the student on its own generated trajectories rather than on teacher-generated ones, thereby directly addressing this mismatch. Underlying virtually all-trajectory-based dense OPD formulations, however, is a foundational premise: that whenever the teacher's next-token logits (unnormalized vocabulary scores) are accessible along a rollout, applying supervision across the entire response is uniformly beneficial.

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.

We argue that this premise is fundamentally flawed, and that on-policy distillation exhibits **another form of prefix drift**, particularly in *strong-to-weak OPD*, where a high-capacity teacher supervises a substantially smaller student (e.g., 30B supervising 1.7B). Specifically, in this regime, the utility of dense supervision depends not merely on whether a global preference gap exists between teacher and student, but on **whether the teacher, given the student's prefix, provides discriminative contrast among the student's next reachable alternative actions**. This mirrors a core principle from outcome-driven reinforcement learning (RL): policy-gradient updates derive their informative learning signal from contrast among competing actions (Qu et al., 2026; Bae et al., 2026; Shu et al., 2026; Yu et al., 2025; Bai et al., 2026), whereas flat feedback across alternatives yields negligible gradient direction. When strong-to-weak OPD supervises rollout regions where teacher contrast has vanished, the student is forced to optimize over a low-contrast, nearly uniform signal, degrading distillation efficiency (Liu et al., 2025a) and impairing out-of-domain (OOD) generalization—performance on tasks held out from the training mixture.

We term this prevalent and trajectory-specific failure mode **local discriminability collapse**: a sustained decline in the teacher's local discriminative contrast along a student rollout. Diagnosing this phenomenon reveals a striking decoupling (Figure 1). Even when the mean teacher–student advantage on sampled tokens remains non-zero late into a response, the dispersion of these advantages and the teacher's margin over competing student alternatives decrease sharply. Consequently, the presence of teacher guidance does not guarantee its local discriminative value for student policy updates.

To resolve this issue, we propose **Local-Discriminability-Aware On-Policy Distillation (LDA-OPD)**, a framework that restricts dense teacher supervision to trajectory regions where teacher feedback remains locally discriminative. At each decoding step, LDA-OPD constructs the student's high-probability candidate set—the most likely next tokens under the student—and computes the teacher's log-probability margin between the top-two candidates in that set. We aggregate this nearest-competitor margin across *sentence segments* (contiguous spans split at sentence boundaries, with a math/code-aware tokenizer) and identify the downward change point using a profiled Bayesian Information Criterion (BIC): a residual-sum-of-squares (RSS) model-selection test in which the noise variance is profiled out of the likelihood. Dense OPD supervision is retained prior to the detected change point and *released* thereafter, meaning the token-level OPD loss is set to zero on the suffix. Crucially, by preserving *loss mass*—rescaling retained-prefix advantages so each sample keeps its original total loss magnitude—LDA-OPD is an adaptive supervision-selection mechanism: in a single rollout pass, without searching a cutoff length, it prunes uninformative suffixes and concentrates learning on high-contrast prefixes. This release applies only to the training loss; inference-time generation remains unconstrained.

Extensive evaluations across student scales (1.7B, 4B, 8B), model families (Qwen3, Gemma3), and diverse task types (multi-step mathematical reasoning, code generation, and out-of-domain scientific reasoning and instruction following) show that LDA-OPD consistently outperforms standard full-trajectory OPD and matches or surpasses multi-run fixed-prefix oracles in a single training run. Matched-framework signal ablations and multi-seed evaluations further indicate that these gains are driven by local action contrast rather than generic entropy or position heuristics.

The contributions of this paper are threefold:

1. **Uncovering local discriminability collapse**: We identify and formalize local discriminability collapse in strong-to-weak OPD, showing that a non-zero global teacher advantage frequently coexists with a severe decay in local action-level contrast.

2. **The LDA-OPD framework and adaptive release**: We introduce LDA-OPD, a lightweight framework (sweep-free over cutoff lengths or static truncation schedules) that measures teacher margins over the student's reachable candidates, aggregates signals across sentence segments, and detects downward change points via profiled RSS-BIC to release low-contrast suffixes.

3. **Empirical and mechanistic validation**: Across student scales (1.7B, 4B, 8B), model families (Qwen3, Gemma3), and diverse task types (multi-step math reasoning, code generation, scientific OOD, and instruction following), we show consistent gains over full OPD. Matched-framework ablations provide strong empirical and counterfactual evidence that the improvements are driven by local action contrast rather than position or entropy heuristics.

2 Related Work

Off-policy distillation. Off-policy distillation addresses distribution mismatch at the trajectory level by modifying the supervision format, data selection, or trajectory transformations. Standard SFT on teacher-generated chain-of-thought traces—step-by-step teacher solutions—is highly sensitive to data provenance and structure, with downstream performance concentrated on actions that originated from the teacher (Liu et al., 2025b). Furthermore, SFT fails to capture sequence-level error recovery and suffers from exposure bias (Yan et al., 2026), while log-probability-based data filtering is frequently confounded by response length (Wang et al., 2026a). Recent methods mitigate these macro-level discrepancies through adversarial data generation loops (Jiang et al., 2023), preference optimization across model architectures (Singh et al., 2025), or difficulty-adaptive trajectory rewrites (Mendes et al., 2026). While these off-policy techniques alleviate trajectory-level distribution shift, securing fine-grained, adaptive token-level supervision remains an open challenge, motivating on-policy distillation when teacher logits are accessible.

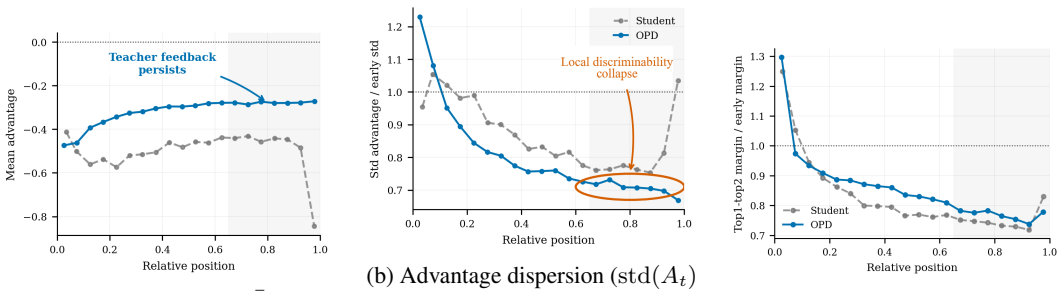
On-policy distillation. On-policy distillation (OPD) trains a student directly on its self-generated rollouts to eliminate exposure bias by evaluating states the student actually visits (Agarwal et al., 2024). Subsequent studies refine the OPD objective rather than the supervision region: G-OPD casts distillation as KL-constrained RL (a policy-gradient loss plus a Kullback–Leibler penalty toward a reference); ExOPD introduces reward extrapolation (Yang et al., 2026), replacing the raw teacher log-probability gap with an extrapolated reward so the student can improve beyond the teacher; and entropy-aware OPD modulates divergence penalties by policy uncertainty (Jin et al., 2026). To address sparse token feedback and vocabulary mismatch, recent analyses motivate top-K support matching (Fu et al., 2026)—aligning teacher and student mass on a shared high-probability token set—or contrast-aware token weighting (Zhang et al., 2026b). Throughout this paper, *support* means that restricted next-token set, not a statistical estimator. To reduce compute, FastOPD restricts OPD to a fixed prefix (Zhang et al., 2026a), using a globally chosen length or a heuristic decay schedule. Our Fixed-prefix OPD baselines instantiate this FastOPD-style protocol.

Recent investigations recognize that dense teacher supervision in OPD is not uniformly reliable across tokens. Uni-OPD (Hou et al., 2026) observes that token-level teacher guidance can contradict overall sequence correctness, and calibrates token margins with the sequence-level outcome to restore global consistency. Concurrent work by Wang et al. (2026b) and CaOPD (Zhang et al., 2026b) study pointwise token learnability (whether a single token's teacher signal is usable) and support compatibility, showing that raw KL divergence conflates learnable on-support mass with off-support divergence on tokens the student cannot reach. Distinct from pointwise token filtering or outcome-centric calibration, our work uncovers a temporal, *state-local* failure mode—a property of the current prefix $(x, y_{<t})$, not of the single sampled token—along autoregressive rollouts: *local discriminability collapse*. This is another form of prefix drift: the student is already conditioning on its own prefixes, yet the teacher can still lose contrast among the next reachable actions. We show that even when global teacher advantage persists, this decay is prevalent and trajectory-specific in strong-to-weak distillation rather than uniform across all suffixes. Unlike fixed-prefix heuristics that sweep a global length, LDA-OPD chooses a per-trajectory cutoff from intrinsic local contrast. OPD has also expanded to multimodal alignment (Bousselham et al., 2025; Li et al., 2026; Cao et al., 2026a) and self-distillation (Chen et al., 2024; Zhao et al., 2026; Sang et al., 2026; Stein et al., 2026; Ding, 2026; Khan et al., 2026); we instead ask a narrower white-box question: when teacher logits are readable, where does dense supervision remain locally discriminative along the student's rollout?

3 Local Discriminability Analysis and the LDA-OPD Paradigm

Preliminaries. In on-policy distillation (OPD), the student policy generates an autoregressive response $y = (y_1, \ldots, y_T)$ conditioned on an input x, and the teacher model evaluates the visited prefix sequence $(x, y_{<t})$ at each step t (Agarwal et al., 2024; Yang et al., 2026; Xiaomi, 2026; Lu & Thinking Machines Lab, 2025). Because evaluation occurs on self-generated rollouts rather than static offline corpora, OPD directly confronts and resolves exposure bias.

Two main objectives exist for OPD: minimizing token-level Kullback-Leibler (KL) divergence over the vocabulary, and policy-gradient optimization via advantage estimation. In this work, we adopt the advantage formulation, which is widely adopted in industrial-scale distillation pipelines (Team, 2025;



(a) Mean advantage ($\bar{A}_t$). (b) Advantage dispersion ($\mathrm{std}(A_t)$ / early). (c) Candidate margin (M_t / early).

Figure 1: **Diagnosing local discriminability collapse and the state-local candidate margin.** (a) Bin-wise mean advantage $\bar{A}_t$ persists late into rollouts (≈ -0.3). (b) Within-bin $\mathrm{std}(A_t)$ divided by an early-position baseline, so 1.0 is the early-rollout level. (c) Teacher nearest-competitor margin M_t on the student's top-K candidates, scaled the same way, decays along the trajectory.

Yang et al., 2026; Xiaomi, 2026; GLM, 2026; Wu et al., 2026; Lu & Thinking Machines Lab, 2025) due to its memory efficiency. Under this formulation, the pointwise teacher-student advantage on the sampled token y_t is defined as:

$$A_t = \log p_T(y_t \mid x, y_{<t}) - \log p_S(y_t \mid x, y_{<t}).$$

The sign and magnitude of A_t reflect the teacher's preference relative to the student on the realized token: $A_t > 0$ means the teacher assigns higher probability to y_t than the student does. Standard full-trajectory OPD applies this advantage across all decoding steps $t \in \{1, \ldots, T\}$ to weight policy gradients.

Diagnosing local discriminability collapse: Silent decay of action contrast. Standard OPD implicitly assumes that dense teacher feedback provides a monotonically informative learning signal throughout the response trajectory. In strong-to-weak distillation—where a high-capacity teacher (e.g., Qwen3-30B-A3B-Instruct-2507, a 30B mixture-of-experts model with 3B active parameters) supervises a smaller student (e.g., Qwen3-1.7B)—this assumption severely breaks down.

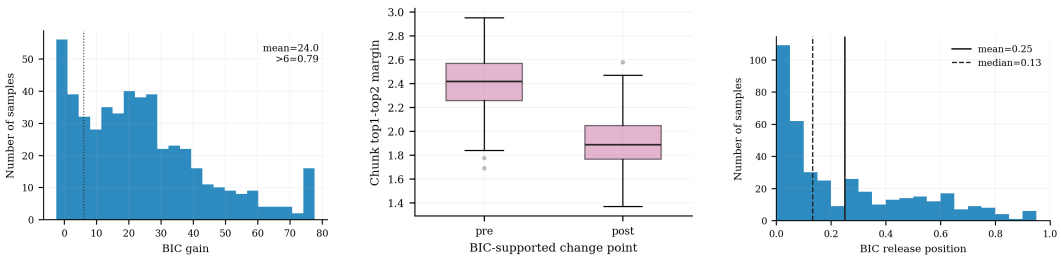
To systematically investigate this breakdown, we examine 500 rollouts from both the base student and its OPD-optimized checkpoint. For each trajectory of length T, we partition tokens by relative position t/T into 20 equal-width bins and compute the bin-wise mean advantage $\bar{A}_t$ and the within-bin dispersion $\mathrm{std}(A_t)$. Figure 1b–c divide the binned dispersion and the candidate margin by an early-position baseline, matching the plotted axes, so that 1.0 marks the early-rollout level rather than a raw unit.

As shown in Figure 1, we uncover a striking decoupling. In Figure 1a, the mean advantage $\bar{A}_t$ remains consistently negative (≈ -0.3) late into the trajectory. However, Figure 1b shows that this early-normalized dispersion declines as the rollout progresses.

In policy gradient learning, informative parameter updates require *action contrast*—the ability to distinguish high-quality actions from sub-optimal alternatives within the current state (Qu et al., 2026; Bae et al., 2026; Shu et al., 2026; Yu et al., 2025; Bai et al., 2026). A flat, unvarying advantage signal across tokens acts as a uniform penalty or reward that provides no directional guidance. We term this phenomenon **local discriminability collapse**: the teacher's local discriminative contrast sharply decays along the rollout despite the persistence of non-zero global advantage. Supervising tokens past this collapse point forces the student to optimize over a low-contrast, nearly uniform signal (Liu et al., 2025a).

From sampled path variance to state-local candidate margin. While the sampled-path statistic $\mathrm{std}(A_t)$ diagnoses the presence of collapse, it is inherently noisy for per-token control because it evaluates only the single realization sampled by the student. Toward the end of a trajectory (e.g., final answer tokens), the teacher may assign a large negative advantage to an incorrect token, creating an artificial spike in single-path variance even though the reasoning path collapsed long before.

To construct a robust, state-local measure of discriminative sharpness—computed at the current prefix rather than from the one sampled token—we ask whether the teacher can discriminate among the



(a) BIC gain distribution. (b) Pre vs. post margin. (c) Release position.

Figure 2: **Empirical properties of LDA-OPD change-point detection.** (a) Profiled RSS-BIC gain ($\mathrm{BIC}_0 - \min_\tau \mathrm{BIC}_1$) exceeds 6 on 79% of trajectories (mean gain 24.0). (b) On accepted trajectories (those with $\mathrm{BIC}_1 < \mathrm{BIC}_0$), mean margin drops sharply across the boundary. (c) Release position, as a fraction of response length, varies widely (mean 0.25, median 0.13), so LDA-OPD is trajectory-specific rather than a static global prefix.

actions the student can take at $(x, y_{<t})$. Let $\mathcal{C}_t$ be the student's top-K next-token set, and write $a_t^{(1)}$ and $a_t^{(2)}$ for the teacher's top two candidates in $\mathcal{C}_t$. The **teacher nearest-competitor margin** is

$$M_t = \log p_T(a_t^{(1)} \mid x, y_{<t}) - \log p_T(a_t^{(2)} \mid x, y_{<t}).$$

Figure 1c shows the same early-normalized decay for M_t. In the unnormalized units of this definition, $M_t \approx 0$ means the teacher is indifferent among reachable alternatives, so the state has lost local discriminative contrast.

Sentence-level aggregation and change-point detection without cutoff sweeps. To filter high-frequency token-level jitter and align supervision boundaries with semantic units, we aggregate M_t over sentence segments identified via custom NLTK tokenization (Wagner, 2010) adapted for mathematical formulas and code blocks (Appendix A.9). For sentence segment $\mathcal{B}_i$,

$$S_i = \log\left(1 + \frac{1}{|\mathcal{B}_i|}\sum_{t\in\mathcal{B}_i} M_t\right).$$

Because $M_t \geq 0$, the $\log(1+\cdot)$ map is well-defined and compresses heavy-tailed margins. We then compare a stable-mean model H_0 with a single downward change-point model $H_1(\tau)$ ($\mu_{\mathrm{pre}} > \mu_{\mathrm{post}}$) using profiled RSS-BIC (Algorithm 1): the Gaussian noise variance is absorbed into RSS/n, with $k_0 = 1$ and $k_1 = 3$. The trajectory is released at $\tau^\star = \arg\min_{\tau:\, \bar{S}_{>\tau} < \bar{S}_{\leq\tau}} \mathrm{BIC}_1(\tau)$ if $\mathrm{BIC}_1(\tau^\star) < \mathrm{BIC}_0$; otherwise it defaults to full OPD. This test is sweep-free over cutoff lengths.

Adaptive supervision release via loss-mass preservation. Once the change point $\tau^\star$ is determined, we construct a binary token mask $q_t \in \{0, 1\}$, retaining tokens up to the end of sentence segment $\tau^\star$. To prevent gradient-magnitude collapse when supervising fewer tokens, we reweight the prefix advantages to preserve the total per-sample loss mass:

$$A'_t = A_t \cdot q_t \cdot \frac{\sum_k l_k}{\sum_k l_k q_k},$$

where $l_k \in \{0, 1\}$ is the standard valid-token loss mask: 1 on response tokens that enter the training loss, and 0 on the prompt and padding. This turns truncation into prefix reweighting: suffix weight is released and concentrated on the high-contrast prefix, without changing the sample-level loss magnitude.

Figure 2 verifies the behavioral properties of LDA-OPD. In Figure 2a, 79% of trajectories have BIC gain above 6 (mean 24.0). A release is accepted only when $\mathrm{BIC}_1 < \mathrm{BIC}_0$; about 8.4% of trajectories (up to $\approx 20\%$ on some task and correctness subsets) never trigger and keep full-trajectory OPD. The contrast between pre- and post-boundary margins verifies the sharpness of the transition, and the wide spread of release positions confirms that LDA-OPD genuinely adapts to individual rollouts rather than mimicking a static global cutoff.

4 EXPERIMENTS

The analytical findings in Section 3 raise fundamental empirical questions: Does selective supervision via LDA-OPD consistently outperform full-trajectory OPD and fixed-prefix baselines? Are the gains driven by local action contrast rather than generic volume or position heuristics? And does LDA-OPD generalize across student capacities, model families, and diverse task paradigms (multi-step mathematical reasoning, code generation, out-of-domain scientific reasoning, and instruction following)?

We investigate these questions in six steps: a main strong-to-weak comparison against full OPD, ExOPD, and multi-run fixed-prefix oracles (Section 4.2); a random-release counterfactual for trajectory coupling (Section 4.3); support-size and $\mathrm{std}(A_t)$ sanity checks (Section 4.4); a matched-framework signal ablation (Section 4.5); a truncation-versus-reweighting isolation (Section 4.6); and scaling to 4B/8B students with Gemma3, GPQA, and IFEval (Section 4.7).

4.1 EXPERIMENTAL SETUP

Our primary evaluation adopts the strong-to-weak OPD protocol (Song & Zheng, 2026; Yang et al., 2026) using Qwen3-1.7B-Non-Thinking as the student and Qwen3-30B-A3B-Instruct-2507 as the teacher. Non-Thinking denotes the Qwen3 variant that answers directly, without emitting an explicit intermediate thinking trace. Training uses the DeepMath-103k prompt set (He et al., 2025), hereafter DeepMath. Evaluation covers diverse task paradigms and domains: five mathematical reasoning benchmarks: AIME25 (Cao et al., 2026b), HMMT25-Feb (Balunovic et al., 2025), HMMT25-Nov (Balunovic et al., 2025), MATH500 (Lightman et al., 2024), and Minerva (Lewkowycz et al., 2022), with AVG denoting their unweighted mean; programmatic code generation across LiveCodeBench v6, HumanEval, and MBPP (Appendix A.8); cross-family distillation on Gemma3; and out-of-domain evaluations spanning scientific QA on GPQA (Rein et al., 2023) and fine-grained instruction following on IFEval (Zhou et al., 2023). Crucially, in all setups, LDA-OPD restricts token-level supervision exclusively during training, preserving the student's full inference-time decoding capability across all benchmark evaluations.

We benchmark the following distillation configurations in Table 1:

- **Student / Teacher**: The base student and the strong reference model.
- **OPD**: Standard on-policy distillation applying dense advantage supervision across all generated tokens (Agarwal et al., 2024).
- **ExOPD**: Reward-extrapolated OPD (Yang et al., 2026). It replaces the raw teacher–student log-probability gap with an extrapolated reward, changing how advantages are computed rather than which tokens are supervised.
- **Fixed-prefix OPD** ($k \in \{1024, 2048, 4096, 8192\}$): A FastOPD-style baseline (Zhang et al., 2026a) that stops dense supervision at a static global token index k. Identifying the best k requires four full training runs and a hindsight choice, so the best of these four is a multi-run oracle. Later tables write FastOPD-2048 for the default $k = 2048$ setting, and FastOPD for that default unless another length is named.
- **Random release**: Keeps LDA-OPD's empirical distribution of release lengths, but randomly reassigns those cutoffs across samples, controlling for supervised-token volume.
- **LDA-OPD (Ours)**: Computes the teacher margin over the student's top-K candidate set, aggregates over sentence segments, and zeros the loss after the detected BIC change point in a single run.

4.2 MAIN RESULTS: ADAPTIVE SUPERVISION VS. FULL OPD

Table 1 demonstrates that while full OPD provides substantial value over the base student (improving AVG from 24.9% to 36.8%), dense full-trajectory supervision is far from optimal. All fixed-prefix baselines outperform full OPD on AVG, corroborating our central insight that low-discriminability suffixes degrade policy optimization.

Crucially, **LDA-OPD achieves 40.1% AVG**, outperforming full OPD by **+3.3%** and surpassing the best retrospectively chosen fixed-prefix oracle (Fixed-prefix-4096 at 39.4%) in a single training run without sweeping cutoff lengths. Furthermore, on out-of-domain GPQA, LDA-OPD achieves 42.4%, significantly exceeding both full OPD (40.7%) and fixed-prefix baselines ($\leq 41.2\%$).

Table 1: Main strong-to-weak OPD results on Qwen3-1.7B. Bold and underline mark the best and second-best trained methods for each metric.

Method	AIME25	HMMT-Feb	HMMT-Nov	MATH500	Minerva	AVG	GPQA
Teacher	61.2%	42.4%	57.9%	98.0%	51.7%	62.2%	70.4%
Student	11.4%	6.6%	4.5%	74.0%	27.9%	24.9%	31.1%
OPD (Agarwal et al., 2024)	28.4%	16.2%	15.9%	86.2%	37.1%	36.8%	40.7%
ExOPD (Yang et al., 2026)	30.8%	16.4%	17.4%	<u>89.2%</u>	**41.2%**	39.0%	36.6%
Fixed-prefix OPD-1024 (Zhang et al., 2026a)	30.1%	18.9%	20.6%	87.8%	35.9%	38.7%	40.7%
Fixed-prefix OPD-2048 (Zhang et al., 2026a)	30.6%	18.5%	<u>20.9%</u>	<u>89.2%</u>	37.3%	39.3%	<u>41.2%</u>
Fixed-prefix OPD-4096 (Zhang et al., 2026a)	**31.6%**	**19.3%**	20.4%	86.8%	<u>38.8%</u>	<u>39.4%</u>	38.6%
Fixed-prefix OPD-8192 (Zhang et al., 2026a)	30.7%	17.4%	18.4%	88.8%	37.3%	38.6%	38.6%
Random release	29.2%	18.0%	18.9%	88.0%	36.0%	38.0%	37.4%
LDA-OPD (Ours)	<u>31.5%</u>	<u>19.1%</u>	**21.9%**	**89.4%**	<u>38.8%</u>	**40.1%**	**42.4%**

Table 2: Support-size and advantage-dispersion checks for LDA-OPD on Qwen3-1.7B. All support-set rows use the fixed teacher top-1/top-2 margin; $\mathrm{std}(A_t)$ applies the same profiled-RSS-BIC procedure.

Variant	AIME25	HMMT-Feb	HMMT-Nov	MATH500	Minerva	AVG	GPQA
OPD (Agarwal et al., 2024)	28.4%	16.2%	15.9%	86.2%	37.1%	36.8%	40.7%
$\mathrm{std}(A_t)$ release	30.3%	17.7%	19.4%	88.4%	38.1%	38.8%	40.9%
$K = 2$ support	30.1%	<u>18.8%</u>	20.8%	87.2%	36.8%	38.7%	<u>42.2%</u>
$K = 4$ support (LDA-OPD)	<u>31.5%</u>	**19.1%**	**21.9%**	**89.4%**	**38.8%**	**40.1%**	**42.4%**
$K = 16$ support	**32.3%**	18.0%	19.4%	<u>88.8%</u>	<u>38.2%</u>	<u>39.3%</u>	41.6%
$K = 32$ support	30.4%	<u>18.8%</u>	<u>20.9%</u>	87.6%	37.5%	39.0%	41.9%

Multi-seed statistical reliability. To verify that these performance gains are statistically sound, Table 8 in Appendix A.9 reports the mean and standard deviation over three independent runs. For Qwen3-1.7B, the three-run AVG is $40.07 \pm 0.12\%$ for LDA-OPD and $38.93 \pm 0.34\%$ for FastOPD-2048. For Qwen3-4B, it is $54.10 \pm 0.16\%$ for LDA-OPD and $52.57 \pm 0.17\%$ for FastOPD-2048. In every run, LDA-OPD strictly dominates the baseline with zero overlap in the reported mean$\pm$std intervals.

4.3 Causal Mechanism: Trajectory Coupling vs. Token Volume

A critical counterfactual hypothesis is whether LDA-OPD's gains arise merely from reducing the total number of supervised tokens rather than identifying trajectory-specific boundaries. The **Random release** baseline decisively refutes this hypothesis: randomly permuting the release points across samples drops AVG performance from 40.1% to 38.0% and degrades GPQA from 42.4% to 37.4%.

This counterfactual provides strong empirical and counterfactual evidence that supervision boundaries must remain tightly coupled to each rollout's local discriminability profile. Figure 2c highlights the wide variance in individual release boundaries, confirming that LDA-OPD performs genuine trajectory-adaptive allocation rather than acting as a static length filter (Wang et al., 2026a).

4.4 Sanity Checks: Support Size and Advantage Dispersion

Table 2 evaluates robustness to the student candidate-set size K (how many top student next-tokens enter $\mathcal{C}_t$) and compares the state-local margin against the realized single-path advantage dispersion $\mathrm{std}(A_t)$.

Across all tested support sizes ($K \in \{2, 4, 16, 32\}$), LDA-OPD consistently maintains strong performance ($\geq 38.7\%$), significantly outperforming full OPD (36.8%). This confirms that K serves as an extremely forgiving, broad local support boundary to capture immediate competing tokens rather than a fragile sample-level hyperparameter. We adopt $K = 4$ as the default because it captures the student's most salient next-token alternatives with minimal computational overhead while achieving the peak score of 40.1%.

Furthermore, while using single-path $\mathrm{std}(A_t)$ improves over full OPD (38.8% vs. 36.8%), it lags behind the state-local margin (40.1%). This confirms the theoretical distinction in Section 3: realized-path variance diagnoses the symptom on a single sample, whereas the candidate-set margin directly measures whether the teacher retains discriminative power over the student's local action space.

4.5 Isolating the Teacher-Margin Signal

To rule out the possibility that the margin signal merely acts as a proxy for generic entropy or sequence position, we perform a controlled ablation. We hold sentence segmentation and the retained-token budget fixed—the latter by matching the mean number of supervised tokens to LDA-OPD—and vary only the release signal. The alternatives are: (i) teacher–student top-K KL, $D_{\mathrm{KL}}(p_T(\cdot \mid \mathcal{C}_t) \,\|\, p_S(\cdot \mid \mathcal{C}_t))$ after both distributions are renormalized on $\mathcal{C}_t$, fed to the same BIC detector; (ii) *Normalized Position (Prefix-25%)*, which bypasses the detector and always releases after the first 25% of tokens, the mean release position of LDA-OPD (Figure 2c), as a fixed relative-length heuristic; (iii) teacher entropy on the candidate set, $H(p_T(\cdot \mid \mathcal{C}_t))$, also fed to the BIC detector.

As shown in Table 3, these alternatives—KL (37.8%), static position (37.4%), and teacher entropy (37.2%)—cluster near full OPD (36.8%). Only the teacher nearest-competitor margin reaches **40.1%**. This provides strong evidence that local action contrast, rather than generic entropy or position heuristics, drives the performance improvements.

Table 3: Matched-framework release signal ablation on Qwen3-1.7B (Avg).

Release Signal	Avg
Teacher Top-1/Top-2 Margin (Ours)	**40.1%**
Teacher–Student Top-K KL	37.8%
Normalized Position (Prefix-25%)	37.4%
Teacher Entropy on Candidate Set	37.2%
Full OPD (No Release)	36.8%

4.6 Truncation versus Loss-Mass Reweighting

LDA-OPD couples suffix truncation with loss-mass preservation, reallocating gradient weight to informative prefixes. To isolate truncation from reweighting, we evaluate an unweighted truncation baseline with *length-normalized loss*: the usual mean over valid tokens, without the $\sum_k l_k / \sum_k l_k q_k$ rescaling (Table 9 in Appendix A.9). On Qwen3-4B, truncation alone achieves 53.1% Avg, already outperforming full OPD (51.6%) and FastOPD-2048 (52.5%). Loss-mass reweighting provides an additional boost to **53.9%**: removing uninformative suffixes (Liu et al., 2025a) delivers the main gain, and rescaling further uses the retained prefix.

4.7 Capacity Scaling, Cross-Family Generality, and OOD Adherence

Table 4: Capacity scaling results with stronger students under Qwen3-30B teacher. Bold and underline denote the best and second-best trained methods within each student scale.

Method	AIME25	HMMT-Feb	HMMT-Nov	MATH500	Minerva	Avg	GPQA
Teacher	61.2%	42.4%	57.9%	98.0%	51.7%	62.2%	70.4%
Student: Qwen3-4B							
Student	19.4%	11.6%	9.2%	83.6%	39.0%	32.5%	48.7%
OPD (Agarwal et al., 2024)	49.9%	30.7%	38.4%	**94.4%**	44.7%	51.6%	<u>55.6%</u>
ExOPD (Yang et al., 2026)	51.8%	30.0%	40.3%	93.4%	**45.6%**	52.2%	43.9%
FastOPD (Zhang et al., 2026a)	<u>52.3%</u>	<u>32.4%</u>	**41.5%**	93.0%	43.4%	<u>52.5%</u>	55.1%
LDA-OPD (Ours)	**54.7%**	**34.9%**	<u>40.7%</u>	<u>94.0%</u>	<u>45.2%</u>	**53.9%**	**57.3%**
Student: Qwen3-8B							
Student	21.6%	11.8%	9.2%	84.6%	42.1%	33.8%	40.2%
OPD (Agarwal et al., 2024)	53.3%	29.3%	42.2%	**95.4%**	45.2%	53.1%	56.8%
ExOPD (Yang et al., 2026)	53.2%	29.1%	**45.8%**	<u>94.0%</u>	**46.3%**	53.7%	39.1%
FastOPD (Zhang et al., 2026a)	**55.3%**	**34.5%**	44.7%	<u>94.0%</u>	45.0%	<u>54.7%</u>	<u>57.8%</u>
LDA-OPD (Ours)	<u>55.1%</u>	<u>33.4%</u>	<u>45.7%</u>	**95.4%**	<u>46.1%</u>	**55.2%**	**58.6%**

To test whether local discriminability collapse persists as the capacity gap—the teacher-minus-student size difference—narrows, we evaluate Qwen3-4B and Qwen3-8B students in Table 4. For 4B, LDA-OPD achieves 53.9% Avg and 57.3% GPQA, outperforming full OPD (51.6% / 55.6%) and FastOPD-2048 (52.5% / 55.1%). For 8B, LDA-OPD achieves 55.2% Avg and 58.6% GPQA. LDA-OPD remains stronger on average as the capacity gap narrows. ExOPD improves in-domain averages

but degrades GPQA across scales because reward extrapolation amplifies the teacher-directed update to pursue performance beyond the teacher without explicitly preserving OOD behavior.

Cross-family generality and instruction following. To confirm that LDA-OPD is not specific to the Qwen architecture, we apply the framework to Gemma3-4B-it → Gemma3-1B-it (Table 5), and additionally report GSM8K (Cobbe et al., 2021), a grade-school math benchmark commonly used for smaller instruction-tuned models. LDA-OPD achieves 62.4% on GSM8K and 46.6% on MATH500, outperforming full OPD (61.1% and 45.8%).

Table 5: Cross-family Gemma3 distillation. GSM8K (Cobbe et al., 2021) is a grade-school math word-problem benchmark.

Method	GSM8K	MATH500
Teacher (Gemma3-4B-it)	88.9%	73.8%
Student (Gemma3-1B-it)	59.6%	44.4%
OPD (Agarwal et al., 2024)	61.1%	45.8%
LDA-OPD (Ours)	**62.4%**	**46.6%**

On OOD IFEval (Zhou et al., 2023) (Table 10 in Appendix A.9), we report the official prompt-level and instruction-level strict accuracies (every constraint in the prompt, or every atomic instruction, must be satisfied). LDA-OPD reaches 68.95% prompt-strict and 76.98% instruction-strict, above FastOPD-2048 (67.47% / 75.90%). The same trend holds for programmatic code generation (Appendix A.8): on LiveCodeBench, HumanEval, and MBPP, LDA-OPD improves upon full OPD and slightly outperforms FastOPD-2048 on average. Selective supervision therefore transfers across the four task types promised above—mathematical reasoning, code generation, scientific OOD, and instruction following—rather than being confined to long-horizon math.

5 Conclusion

In this work, we asked a concrete question in strong-to-weak on-policy distillation: *when and where should teachers stop teaching?* We formalized **local discriminability collapse**: a non-zero global teacher–student advantage can coexist with a decay in local action-level contrast, so full-trajectory OPD optimizes over a low-contrast, nearly uniform suffix signal. **LDA-OPD** measures state-local candidate margins, aggregates them over sentence segments, and uses profiled RSS-BIC to release—that is, to stop applying—dense loss on the low-contrast suffix while preserving per-sample loss mass. Across multi-step mathematical reasoning, code generation, out-of-domain scientific reasoning, and instruction following, as well as student scales and model families, LDA-OPD outperforms standard full-trajectory OPD, matches or exceeds multi-run fixed-prefix oracles in a single training run without sweeping cutoff lengths, and provides strong empirical and counterfactual evidence attributing the gain to local action contrast. Crucially, because LDA-OPD operates strictly as a training-time supervision selection mechanism, the student retains full unconstrained generation capacity at inference. Future alignment methods should evaluate not only whether teacher guidance is present, but whether it remains locally sharp.

AI Use Statement

In this work, we used generative AI tools to edit the manuscript for grammar, wording, and LaTeX organization.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Jacob Austin, Augustus Odena, Maxwell I. Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie J. Cai, Michael Terry, Quoc V. Le, and Charles Sutton. Program synthesis

with large language models. *CoRR*, abs/2108.07732, 2021. URL https://arxiv.org/abs/2108.07732.

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, Jeongyeon Nam, and Donghyun Kwak. Online difficulty filtering for reasoning oriented reinforcement learning. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2026 - Volume 1: Long Papers, Rabat, Morocco, March 24-29, 2026*, pp. 700–719. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.EACL-LONG.30. URL https://doi.org/10.18653/v1/2026.eacl-long.30.

Yang Bai, Kaiyuan Liu, Ziyuan Zhuang, Jiahong Zhou, Rongxiang Weng, Xin Chen, Jingang Wang, and Xunliang Cai. Multi-objective and mixed-reward reinforcement learning via reward-decorrelated policy optimization. *CoRR*, abs/2605.13641, 2026. doi: 10.48550/ARXIV.2605.13641. URL https://doi.org/10.48550/arXiv.2605.13641.

Mislav Balunovic, Jasper Dekoninck, Ivo Petrov, Nikola Jovanovic, and Martin T. Vechev. Matharena: Evaluating llms on uncontaminated math competitions. 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/1d27c01ebd3e3aebe226b44fc970d803-Abstract-Datasets_and_Benchmarks_Track.html.

Walid Bousselham, Hilde Kuehne, and Cordelia Schmid. VOLD: reasoning transfer from llms to vision-language models via on-policy distillation. *CoRR*, abs/2510.23497, 2025. doi: 10.48550/ARXIV.2510.23497. URL https://doi.org/10.48550/arXiv.2510.23497.

Di Cao, Dongjie Fu, Hai Yu, Siqi Zheng, Xu Tan, and Tao Jin. X-OPD: cross-modal on-policy distillation for capability alignment in speech llms. *CoRR*, abs/2603.24596, 2026a. doi: 10.48550/ARXIV.2603.24596. URL https://doi.org/10.48550/arXiv.2603.24596.

Maosong Cao, Kai Chen, Haodong Duan, Yixiao Fang, Zhiwei Fei, Tong Gao, Ge Jiaye, Mo Li, Hongwei Liu, Junnan Liu, Yuan Liu, Chengqi Lyu, Han Lyu, Ningsheng Ma, Zerun Ma, Yu Sun, Zhiyong Wu, Linchen Xiao, Jun Xu, Haochen Ye, Zhaohui Yu, Yike Yuan, Songyang Zhang, Yufeng Zhao, Fengzhe Zhou, Peiheng Zhou, Dongsheng Zhu, Lin Zhu, and Jingming Zhuo. Opencompass: A universal evaluation platform for large language models, 2026b. URL https://doi.org/10.48550/arXiv.2605.19276.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. 235:6621–6642, 2024. URL https://proceedings.mlr.press/v235/chen24j.html.

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

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong

Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=9SkkifLopZ.

Ken Ding. HDPO: hybrid distillation policy optimization via privileged self-distillation. *CoRR*, abs/2603.23871, 2026. doi: 10.48550/ARXIV.2603.23871. URL https://doi.org/10.48550/arXiv.2603.23871.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *CoRR*, abs/2603.25562, 2026. doi: 10.48550/ARXIV.2603.25562. URL https://doi.org/10.48550/arXiv.2603.25562.

GLM. GLM-5: from vibe coding to agentic engineering. *CoRR*, abs/2602.15763, 2026. doi: 10.48550/ARXIV.2602.15763. URL https://doi.org/10.48550/arXiv.2602.15763.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *CoRR*, abs/2504.11456, 2025. doi: 10.48550/ARXIV.2504.11456. URL https://doi.org/10.48550/arXiv.2504.11456.

Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, Chengquan Zhang, Zhuotao Tian, Han Hu, Yi Yang, Fei Wu, and Hehe Fan. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *CoRR*, abs/2605.03677, 2026. doi: 10.48550/ARXIV.2605.03677. URL https://doi.org/10.48550/arXiv.2605.03677.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. Lion: Adversarial distillation of proprietary large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 3134–3154. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.189. URL https://doi.org/10.18653/v1/2023.emnlp-main.189.

Woogyeol Jin, Taywon Min, Yongjin Yang, Swanand Ravindra Kadhe, Yi Zhou, Dennis Wei, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *CoRR*, abs/2603.07079, 2026. doi: 10.48550/ARXIV.2603.07079. URL https://doi.org/10.48550/arXiv.2603.07079.

Rana Muhammad Shahroz Khan, Zijie Liu, Zhen Tan, Charles Fleming, and Tianlong Chen. TMS: trajectory-mixed supervision for reward-free, on-policy SFT. *CoRR*, abs/2602.03073, 2026. doi: 10.48550/ARXIV.2602.03073. URL https://doi.org/10.48550/arXiv.2602.03073.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html.

Jiaze Li, Hao Yin, Haoran Xu, Boshen Xu, Wenhui Tan, Zewen He, Jianzhong Ju, Zhenbo Luo, and Jian Luan. Video-opd: Efficient post-training of multimodal large language models for temporal video grounding via on-policy distillation. *CoRR*, abs/2602.02994, 2026. doi: 10.48550/ARXIV.2602.02994. URL https://doi.org/10.48550/arXiv.2602.02994.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Kaiyuan Liu, Chen Shen, Zhanwei Zhang, Junjie Liu, Xiaosong Yuan, and Jieping Ye. Efficient reasoning through suppression of self-affirmation reflections in large reasoning models. *CoRR*, abs/2506.12353, 2025a. doi: 10.48550/ARXIV.2506.12353. URL https://doi.org/10.48550/arXiv.2506.12353.

Kaiyuan Liu, Shaotian Yan, Rui Miao, Bing Wang, Chen Shen, Jun Zhang, and Jieping Ye. Where did this sentence come from? tracing provenance in LLM reasoning distillation. *CoRR*, abs/2512.20908, 2025b. doi: 10.48550/ARXIV.2512.20908. URL https://doi.org/10.48550/arXiv.2512.20908.

Kevin Lu and Thinking Machines Lab. On-policy distillation, 2025. URL https://thinkingmachines.ai/blog/on-policy-distillation. Connectionism.

Ethan Mendes, Jungsoo Park, and Alan Ritter. Didactic to constructive: Turning expert solutions into learnable reasoning. *CoRR*, abs/2602.02405, 2026. doi: 10.48550/ARXIV.2602.02405. URL https://doi.org/10.48550/arXiv.2602.02405.

Yun Qu, Qi Wang, Yixiu Mao, Vincent Tao Hu, Björn Ommer, and Xiangyang Ji. Can prompt difficulty be online predicted for accelerating RL finetuning of reasoning models? In Srinivasan Parthasarathy, David F. Gleich, Xiangliang Zhang, Wee Hyong Tok, Faisal Farooq, Qi He, Ambuj K. Singh, Haixun Wang, and Yan Liu (eds.), *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1, KDD 2026, Jeju Island, Korea, August 9-13, 2026*, pp. 1240–1250. ACM, 2026. doi: 10.1145/3770854.3780263. URL https://doi.org/10.1145/3770854.3780263.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. volume abs/2311.12022, 2023. doi: 10.48550/ARXIV.2311.12022. URL https://doi.org/10.48550/arXiv.2311.12022.

Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. Crisp: Compressed reasoning via iterative self-policy distillation. *arXiv preprint arXiv:2603.05433*, 2026.

Dong Shu, Denghui Zhang, and Jessica Hullman. Learning from the right rollouts: Data attribution for ppo-based LLM post-training. *CoRR*, abs/2604.01597, 2026. doi: 10.48550/ARXIV.2604.01597. URL https://doi.org/10.48550/arXiv.2604.01597.

Aasheesh Singh, Vishal Vaddina, and Dagnachew Birru. Orpo-distill: Mixed-policy preference optimization for cross-architecture LLM distillation. *CoRR*, abs/2509.25100, 2025. doi: 10.48550/ARXIV.2509.25100. URL https://doi.org/10.48550/arXiv.2509.25100.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *CoRR*, abs/2604.00626, 2026. doi: 10.48550/ARXIV.2604.00626. URL https://doi.org/10.48550/arXiv.2604.00626.

Alex Stein, Furong Huang, and Tom Goldstein. GATES: self-distillation under privileged context with consensus gating. *CoRR*, abs/2602.20574, 2026. doi: 10.48550/ARXIV.2602.20574. URL https://doi.org/10.48550/arXiv.2602.20574.

Qwen Team. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025. doi: 10.48550/ARXIV.2505.09388. URL https://doi.org/10.48550/arXiv.2505.09388.

Wiebke Wagner. *Steven Bird, Ewan Klein and Edward Loper: Natural Language Processing with Python, Analyzing Text with the Natural Language Toolkit - O'Reilly Media, Beijing, 2009, ISBN 978-0-596-51649-9*, volume 44. 2010. doi: 10.1007/S10579-010-9124-X. URL https://doi.org/10.1007/s10579-010-9124-x.

Bing Wang, Rui Miao, Chen Shen, Shaotian Yan, Kaiyuan Liu, Ximing Li, Xiaosong Yuan, Sinan Fan, Jun Zhang, and Jieping Ye. On the step length confounding in LLM reasoning data selection. pp. 18443–18457, 2026a. doi: 10.18653/V1/2026.FINDINGS-ACL.918. URL https://doi.org/10.18653/v1/2026.findings-acl.918.

Yuanyi Wang, Su Lu, Yanggan Gu, Pengkai Wang, Yifan Yang, Zhaoyi Yan, Congkai Xie, Jianmin Wu, and Hongxia Yang. Not all disagreement is learnable: Token teachability in on-policy distillation. *CoRR*, abs/2605.26844, 2026b. doi: 10.48550/ARXIV.2605.26844. URL https://doi.org/10.48550/arXiv.2605.26844.

Yecheng Wu, Song Han, and Hai Cai. Lightning OPD: efficient post-training for large reasoning models with offline on-policy distillation. *CoRR*, abs/2604.13010, 2026. doi: 10.48550/ARXIV.2604.13010. URL https://doi.org/10.48550/arXiv.2604.13010.

LLM-Core Xiaomi. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

Shaotian Yan, Kaiyuan Liu, Chen Shen, Bing Wang, Sinan Fan, Jun Zhang, Yue Wu, Zheng Wang, and Jieping Ye. Distribution-aligned sequence distillation for superior long-cot reasoning. *CoRR*, abs/2601.09088, 2026. doi: 10.48550/ARXIV.2601.09088. URL https://doi.org/10.48550/arXiv.2601.09088.

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *CoRR*, abs/2602.12125, 2026. doi: 10.48550/ARXIV.2602.12125. URL https://doi.org/10.48550/arXiv.2602.12125.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Dongxu Zhang, Zhichao Yang, Sepehr Janghorbani, Jun Han, Andrew Ressler II, Qian Qian, Gregory D. Lyng, Sanjit Singh Batra, and Robert E. Tillman. Fast and effective on-policy distillation from reasoning prefixes. pp. 25553–25569, 2026a. doi: 10.18653/V1/2026.FINDINGS-ACL.1276. URL https://doi.org/10.18653/v1/2026.findings-acl.1276.

Jiaxin Zhang, Xiangyu Peng, Qinglin Chen, Qinyuan Ye, Caiming Xiong, and Chien-Sheng Wu. The illusion of certainty: Decoupling capability and calibration in on-policy distillation. *CoRR*, abs/2604.16830, 2026b. doi: 10.48550/ARXIV.2604.16830. URL https://doi.org/10.48550/arXiv.2604.16830.

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. *CoRR*, abs/2601.18734, 2026. doi: 10.48550/ARXIV.2601.18734. URL https://doi.org/10.48550/arXiv.2601.18734.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. volume abs/2311.07911, 2023. doi: 10.48550/ARXIV.2311.07911. URL https://doi.org/10.48550/arXiv.2311.07911.

A Technical Appendices and Supplementary Material

A.1 Algorithm Details of LDA-OPD

The algorithm below summarizes the LDA-OPD pipeline used throughout our experiments and expands the profiled RSS-BIC test stated in Section 3. The reference implementation historically called the same procedure *dynamic-prefix OPD*; that name is an implementation alias, not a second method. The pipeline: (1) extracts the student's top-K candidate set, (2) computes the nearest-competitor teacher margin, (3) aggregates margins across sentence segments, (4) runs the profiled RSS-BIC change-point test, and (5) applies loss-mass-preserving prefix reweighting.

Algorithm 1 Local-Discriminability-Aware On-Policy Distillation (LDA-OPD)

Require: Response tokens $y_{1:T}$, token advantages $A_{1:T}$, loss mask $l_{1:T}$, teacher probabilities on student top-K candidates, NLTK sentence segments $\mathcal{B}_1, \ldots, \mathcal{B}_n$
Ensure: Truncated and rescaled advantages $A'_{1:T}$

for each response token position t **do**
  $\mathcal{C}_t \leftarrow \mathrm{TopK}_a\ p_S(a \mid x, y_{<t})$
  Rank candidates in $\mathcal{C}_t$ by teacher log-probability
  $a_t^{(1)} \leftarrow \arg\max_{a \in \mathcal{C}_t} \log p_T(a \mid x, y_{<t})$
  $a_t^{(2)} \leftarrow \arg\max_{a \in \mathcal{C}_t \setminus \{a_t^{(1)}\}} \log p_T(a \mid x, y_{<t})$
  $M_t \leftarrow \log p_T(a_t^{(1)} \mid x, y_{<t}) - \log p_T(a_t^{(2)} \mid x, y_{<t})$
end for
for each sentence segment $\mathcal{B}_i$ **do**
  $S_i \leftarrow \log\Big(1 + \frac{1}{|\mathcal{B}_i|} \sum_{t \in \mathcal{B}_i} M_t\Big)$
end for
$\mathrm{BIC}_0 \leftarrow n \log((\sum_i (S_i - \bar{S})^2 + \epsilon)/n) + \log n$
$\mathcal{T} \leftarrow \{\tau \in \{1, \ldots, n-1\} : \bar{S}_{>\tau} < \bar{S}_{\leq\tau}\}$
for each $\tau \in \mathcal{T}$ **do**
  $\mathrm{RSS}_1(\tau) \leftarrow \sum_{i \leq \tau} (S_i - \bar{S}_{\leq\tau})^2 + \sum_{i > \tau} (S_i - \bar{S}_{>\tau})^2$
  $\mathrm{BIC}_1(\tau) \leftarrow n \log((\mathrm{RSS}_1(\tau) + \epsilon)/n) + 3 \log n$
end for
$\tau^\star \leftarrow \arg\min_{\tau \in \mathcal{T}} \mathrm{BIC}_1(\tau)$
if $\mathcal{T} \neq \emptyset$ and $\mathrm{BIC}_1(\tau^\star) < \mathrm{BIC}_0$ **then**
  $q_t \leftarrow 1$ for tokens before the boundary after segment $\tau^\star$, and $q_t \leftarrow 0$ otherwise
else
  $q_t \leftarrow 1$ for all valid response tokens
end if
$A'_t \leftarrow A_t\, q_t\, \frac{\sum_t l_t}{\max(\sum_t l_t q_t, \epsilon)}$
return $A'_{1:T}$

A.2 Theoretical Motivation for Local Discriminability

This section motivates the release rule. We use *local teachability* only as a synonym for local discriminability—whether the teacher still distinguishes among actions the student can take—not as a second quantity. The goal is not to prove that BIC always recovers a true change point, nor that truncation always reduces variance. We formalize why local discriminability of teacher feedback matters for OPD, and why a penalized downward-change rule is a statistically meaningful way to use this signal.

Two notions of advantage variation. The analysis in Section 3 uses the dispersion of sampled-token advantages,

$$A_t(y_t) = \log p_T(y_t \mid x, y_{<t}) - \log p_S(y_t \mid x, y_{<t}),$$

along a realized trajectory. This statistic is a diagnostic symptom: it shows that dense OPD feedback can lose token-specific variation as the response progresses. The release rule does not directly use this trajectory-wise $\mathrm{std}(A_t(y_t))$. Instead, it uses a state-local proxy computed at a fixed prefix $h_t = (x, y_{<t})$: whether the teacher can still distinguish among the student's likely next-token candidates. The derivation below is about this second object, namely action-wise variation inside

the student's local candidate set $\mathcal{C}_t$. Thus, sampled-path advantage dispersion motivates the problem, while the top-K teacher margin operationalizes the release decision.

Local teacher signal and policy-gradient usefulness. For a candidate next token a at a fixed prefix h_t, define the action-wise OPD advantage

$$A_t(a) = \log p_T(a \mid h_t) - \log p_S(a \mid h_t),$$

and the score vector

$$s_t(a) = \nabla_\theta \log p_S(a \mid h_t).$$

The local expected OPD update over a candidate distribution π_t supported on the student's reachable set $\mathcal{C}_t$ is

$$g_t = \mathbb{E}_{a\sim\pi_t}\left[A_t(a)s_t(a)\right].$$

This is an action-wise analysis at one prefix, not a temporal statistic over adjacent sampled tokens. In the standard policy-gradient baseline argument, if the expectation is taken under the full student distribution, then $\mathbb{E}_{a\sim p_S(\cdot|h_t)}[s_t(a)] = 0$, so subtracting any action-independent baseline from $A_t(a)$ does not change the expected update. For the restricted top-K set, this identity is only approximate because π_t is a locally renormalized proxy; equivalently, we can view the following centered expression as the informative component that removes action-independent offsets:

$$\tilde{g}_t \approx \mathbb{E}_{a\sim\pi_t}\left[(A_t(a) - \bar{A}_t)s_t(a)\right], \qquad \bar{A}_t = \mathbb{E}_{a\sim\pi_t}[A_t(a)].$$

By Cauchy's inequality,

$$\|\tilde{g}_t\|^2 \le \mathrm{Var}_{a\sim\pi_t}(A_t(a)) \cdot \mathbb{E}_{a\sim\pi_t}\|s_t(a)\|^2.$$

This bound explains the central distinction made in the paper. A nonzero sampled-token advantage along a trajectory does not by itself imply a useful dense supervision signal. What matters for local teaching is whether the teacher creates action-dependent contrast among alternatives the student is likely to take. If the teacher's preference over the student's reachable alternatives becomes locally flat, then the action-dependent component of the update becomes small, and dense token-level feedback becomes less informative.

Why teacher margin is a proxy for local discriminability. One could also compute $\mathrm{Var}_{\mathcal{C}_t}(A_t(a))$ directly over the same top-K candidate set. This would measure action-wise advantage dispersion within the local support, and its computational cost is comparable when teacher and student scores for all top-K candidates are already available. We use the teacher's nearest-competitor margin instead because it targets a narrower decision question: whether the teacher has a clear first choice over the actions the student is most likely to take. The margin is defined inside the student's top-K candidate set as

$$M_t = \log p_T(a_t^{(1)} \mid h_t) - \log p_T(a_t^{(2)} \mid h_t),$$

where $a_t^{(1)}$ and $a_t^{(2)}$ are the teacher's top two candidates within $\mathcal{C}_t$. This quantity does not fully characterize the local advantage distribution. Instead, it measures a narrower but useful property: whether the teacher can still express a sharp preference among actions the student is likely to take. A sustained decrease in M_t therefore indicates that teacher feedback may remain globally different from the student while becoming less locally actionable.

Signal-to-noise (SNR) view of release. Let P denote the prefix retained by the release rule and R denote the released suffix. The segment-level signal is the action-dependent component $\tilde{g}_t$, whose magnitude is controlled by local advantage variation in the preceding bound. For a fixed update direction u, define the directional signal and variance

$$m_P = u^\top \mathbb{E}\left[\sum_{t\in P} \tilde{g}_t\right], \qquad v_P = \mathrm{Var}\left(u^\top \sum_{t\in P} \tilde{g}_t\right),$$

and analogously m_R, v_R for the suffix. The full OPD update has directional SNR

$$\mathrm{SNR}_{\mathrm{full}} = \frac{(m_P + m_R)^2}{v_P + v_R},$$

whereas the released update, up to the loss-mass rescaling used in the paper, has the same directional SNR as the retained prefix:

$$\mathrm{SNR}_{\mathrm{release}} = \frac{m_P^2}{v_P}.$$

Therefore release improves directional SNR whenever

$$\frac{m_P^2}{v_P} \geq \frac{(m_P + m_R)^2}{v_P + v_R},$$

or equivalently,

$$\frac{v_R}{v_P} \geq 2\frac{m_R}{m_P} + \left(\frac{m_R}{m_P}\right)^2.$$

This condition captures the intended regime of local discriminability collapse: the suffix contributes little aligned teacher signal relative to its variance. Continuing dense OPD on that suffix can lower the effective SNR of the update. The release rule is designed to detect this regime from the teacher-margin curve.

Role of the BIC-style criterion. The segment scores S_i are noisy summaries of local teacher contrast. The BIC-style rule compares a no-change model,

$$S_i = \mu + \epsilon_i,$$

against a one-drop model,

$$S_i = \begin{cases} \mu_{\mathrm{pre}} + \epsilon_i, & i \leq \tau, \\ \mu_{\mathrm{post}} + \epsilon_i, & i > \tau, \end{cases} \qquad \mu_{\mathrm{pre}} > \mu_{\mathrm{post}}.$$

The profiled RSS-BIC objective penalizes the additional parameters of the one-drop model:

$$\mathrm{BIC}_0 = n \log\left(\frac{\mathrm{RSS}_0 + \epsilon}{n}\right) + \log n,$$

$$\mathrm{BIC}_1(\tau) = n \log\left(\frac{\mathrm{RSS}_1(\tau) + \epsilon}{n}\right) + 3 \log n.$$

Thus, a release is accepted only when the reduction in residual error is large enough to overcome the complexity penalty. This does not guarantee recovery of a true latent change point for every finite trajectory. Rather, it makes the rule statistically disciplined: it avoids releasing merely because of small random fluctuations, while still allowing each trajectory to choose its own supervision boundary.

Interpretation. The analysis supports the main claim: dense OPD should depend not only on whether a teacher–student gap exists, but also on whether the teacher's feedback remains locally discriminative over actions the student can take. The release rule uses the state-local teacher margin as a discriminability proxy and a penalized downward-change criterion to decide when dense supervision should stop.

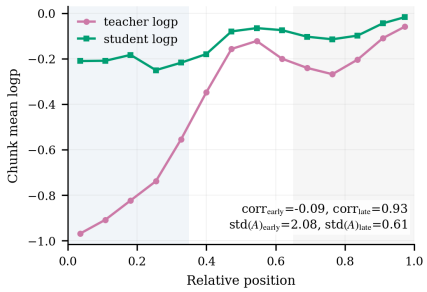


Figure 3: **Qualitative Example**: A representative rollout shows that while teacher and student log-probabilities diverge early on, they become highly aligned in later stages (log-probabilities correlation increases from -0.09 to 0.93). The resulting reduction in advantage variation (std(A) drops from 2.08 to 0.61) renders the dense feedback less actionable for guiding specific local policy improvements.

A.3 Additional Diagnostic

Figure 3 illustrates this symptom. Early in the rollout, the teacher and student log-probability curves diverge substantially, yielding highly variable

advantage values. Yet, as the generation prefix $(x, y_{<t})$ lengthens, the predictions of both models become heavily conditioned on the same accumulated context. Their sampled-token log-probability trajectories align, the standard deviation of the advantages drops, and the scalar feedback becomes less informative as a token-specific priority signal.

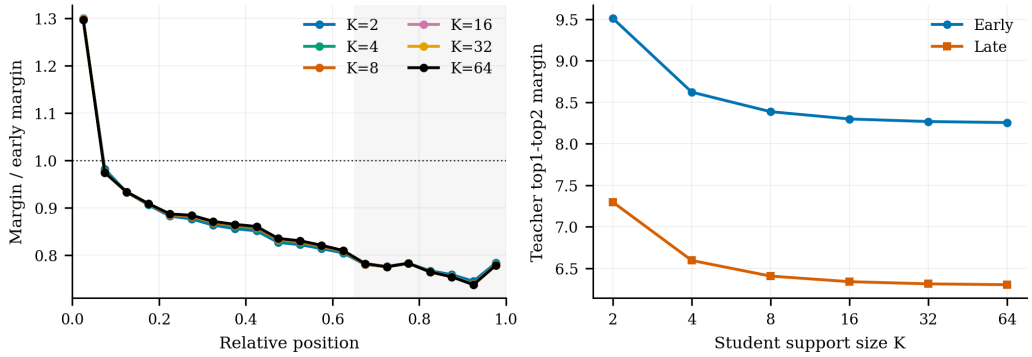


Figure 4: **Support-size sensitivity.** Left: Teacher-margin curves, each divided by an early-position baseline so that 1.0 is the early-rollout level, remain similar across $K \in \{2, 4, 8, 16, 32, 64\}$. Right: Early segments have larger teacher top-1–top-2 margins than late segments for every K, so the decline in local discriminability is not an artifact of one support size.

Figure 4 checks whether the normalized margin decline depends strongly on the chosen support size. The curves remain similar across $K \in \{2, 4, 8, 16, 32, 64\}$, which supports the interpretation that the observed decay is a property of teacher ranking sharpness over student-reachable actions rather than an artifact of one support size. The candidate set size K thus acts as an extremely forgiving neighborhood parameter rather than a delicate or fragile sample-level tuning knob. The main experiments fix the margin definition to top1–top2 and report the training-level support-size check in Table 2.

Figure 5 shows how the BIC-style rule behaves on one trajectory. The accepted release point is not meant to prove that the subsequent tokens are useless; rather, it marks where the aggregated margin curve is better summarized by a lower-contrast post-change segment. This example complements the distribution-level evidence in the main text by showing the object on which the rule operates. The same implementation is applied to all rollouts, with no benchmark-specific cutoff or per-sample manual adjustment. If the fixed criterion does not accept a downward change point, the trajectory falls back to full OPD supervision.

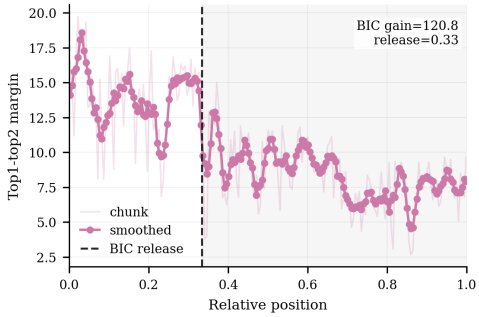


Figure 5: **Example BIC-style release.** A representative rollout shows an accepted downward change point in the teacher top1–top2 margin over the student's top-K candidates.

Figure 6 illustrates the relationship between fixed-prefix checkpoints and the diagnostic release lengths estimated from their respective rollouts. Each data point represents a checkpoint trained with a specific global prefix cutoff. The horizontal axis denotes the mean diagnostic release length, calculated by applying our teacher-margin release rule to rollouts from that checkpoint. The vertical axis represents the checkpoint's performance averaged across five downstream math benchmarks. Consequently, this visualization evaluates whether a checkpoint's rollout distribution maintains locally teachable regions throughout the response, rather than assessing the choice of prefix length in isolation.

This trend suggests a mechanistic interpretation. Better fixed-prefix checkpoints tend to have rollouts whose margin curves imply later diagnostic release lengths, meaning that the teacher remains locally

discriminative for a longer portion of those trajectories. A good fixed prefix is therefore not good merely because a particular cutoff index is special. It works when that global cutoff happens to retain a region where the teacher still provides useful local contrast. Conversely, if the local-discriminability curve collapses early, a larger fixed prefix may add low-value suffix tokens. With only four fixed-prefix checkpoints, this plot is descriptive rather than a correlation claim. It illustrates why a single nominal prefix length is incomplete: the useful boundary depends on the trajectory-level teacher signal as well as the global cutoff.

A.4 Release-Point Case Studies

The following case studies illustrate the specific behaviors that the release rule is designed to exclude from supervision. These instances do not constitute inference-time truncation. Because the trajectory-specific rule exclusively modifies the training objective, the model remains capable of generating complete responses during evaluation.

While the blue regions are not guaranteed to be error-free, they encompass the primary reasoning or setup preceding the release point. In contrast, the red regions highlight repetitive or noisy continuations that would otherwise receive potentially counterproductive dense supervision from the teacher model. These examples thus demonstrate why dense OPD supervision on terminal regions can be of limited utility.

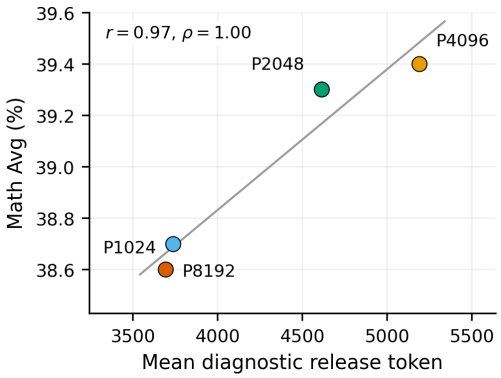


Figure 6: **Diagnostic release length and fixed-prefix performance.** Each point is one fixed-prefix checkpoint. The horizontal axis is mean diagnostic release length. With only four checkpoints, this is descriptive rather than a correlation claim.

Case 1: repeated low-value continuation after the release point. The first example asks for $\lim_{n\to\infty} n^3 x^n (1-x)^4$ for $x \in [0,1]$. The pre-release prefix already contains the correct case split and a ratio argument showing that $a_{n+1}/a_n \to x < 1$, so $a_n \to 0$ for $0 < x < 1$. After the BIC release point, the continuation repeatedly restates the same conclusion through the standard fact that exponential decay dominates polynomial growth, and includes several false starts (Liu et al., 2025a).

Before release: core setup before noisy continuation

We are asked to evaluate the limit:

$$\lim_{n\to\infty} n^3 x^n (1-x)^4 \quad \text{for } x \in [0,1].$$

We will analyze this limit by considering different cases based on the value of x in the interval $[0,1]$.
—
Step 1: Understanding the Behavior
Let's denote:

$$a_n = n^3 x^n (1-x)^4$$

We are to evaluate:

$$\lim_{n\to\infty} a_n$$

We'll analyze this limit depending on whether $x < 1$, $x = 1$, or $x = 0$.
—
Case 1: $x = 0$
Then $x^n = 0^n = 0$ for $n \geq 1$, so:

$$a_n = n^3 \cdot 0 = 0 \quad \text{for all } n \geq 1$$

Therefore:

$$\lim_{n\to\infty} a_n = 0$$

—
Case 2: $x = 1$
Then $x^n = 1^n = 1$, so:

$$a_n = n^3 \cdot 1 \cdot (1-1)^4 = n^3 \cdot 1 \cdot 0 = 0$$

So again:

$$\lim_{n\to\infty} a_n = 0$$

—

Case 3: $0 < x < 1$
Now $x \in (0, 1)$, so $x^n \to 0$ as $n \to \infty$, but n^3 grows to infinity. We need to understand the **rate** at which x^n decays.
Recall that for fixed $x \in (0, 1)$, $x^n \to 0$ exponentially fast.
We can compare $n^3 x^n$ with its exponential decay.
Let's consider the ratio:

$$\frac{a_{n+1}}{a_n} = \frac{(n+1)^3 x^{n+1}(1-x)^4}{n^3 x^n (1-x)^4} = \left(\frac{n+1}{n}\right)^3 x$$
$$= \left(1 + \frac{1}{n}\right)^3 x$$

As $n \to \infty$, $\left(1 + \frac{1}{n}\right)^3 \to 1$, so the ratio $\frac{a_{n+1}}{a_n} \to x$. Since $x \in (0, 1)$, this ratio approaches a value less than 1.
This suggests that $a_n \to 0$, as the sequence a_n is decreasing and bounded below by 0 (since $x > 0$, $x^n > 0$), and the ratio is less than 1.
But let's be more precise.

After release: low-value content

We can use the **standard result**:
For $0 < x < 1$, the sequence $n^k x^n \to 0$ as $n \to \infty$ for any fixed $k \in \mathbb{R}$.
This is a well-known fact in analysis: exponential decay dominates polynomial growth.
In fact, we can apply the **exponential decay** bound:

$$x^n \le e^{-n \cdot \frac{\ln x}{1}} \quad \text{since } x = e^{\ln x} \text{ and } \ln x < 0 \Rightarrow -\ln x > 0$$

But let's use a more precise method.
Let's use the **inequality**:

$$\forall n \ge 1, \quad x^n \le \frac{1}{n^k} \quad ?$$

No, that's not helpful.
Instead, we use:

$$\lim_{n \to \infty} n^3 x^n (1-x)^4 = 0 \quad \text{for } 0 < x < 1$$

Why?
Because $x^n \to 0$ exponentially, while n^3 grows polynomially. The **exponential decay dominates polynomial growth**, so the product goes to 0.
Let's make a more formal argument.
Let $x \in (0, 1)$. Then $x^n \to 0$, and n^3 grows, but we can write:

$$n^3 x^n = n^3 e^{n \ln x} = e^{n \ln n^3 + n \ln x} = e^{n(\ln n^3 + \ln x)}$$

So:

$$\lim_{n \to \infty} n^3 x^n = \lim_{n \to \infty} e^{n(\ln n^3 + \ln x)} = \infty \cdot e^{-\infty} \quad ?$$

Wait — no, wait.
Actually, $n \ln n^3 = 3n \ln n$, and $n \ln x$ is negative since $x < 1$, so $n \ln x \to -\infty$.
So:

$$n^3 x^n = e^{n \ln n^3 + n \ln x} = e^{n(3 \ln n + \ln x)}$$

We analyze the exponent:

$$n(3 \ln n + \ln x) \to \infty \text{ because } n \ln n \to \infty \text{ and } \ln x < 0, \text{ so } n \ln x \to -\infty \text{ but much slower than } n \ln n$$

So overall, $n^3 x^n \to \infty \cdot e^{-\infty}$ — is that 0?
Actually, $n \ln n \to \infty$, and $n \ln x \to -\infty$, but the exponent grows to ∞ because $n \ln n \to \infty$, so $n^3 x^n \to 0$ **exponentially faster than any polynomial growth**.
Wait — let's make a **comparison**:
We can use:

$$n^3 x^n = e^{n \ln n^3 + n \ln x} = e^{n(3 \ln n + \ln x)}$$

Let $a_n = n^3 x^n$, so $\ln a_n = n(3 \ln n + \ln x)$
Now analyze $\ln a_n$:

- $3 \ln n \to \infty$
- $\ln x < 0$, so $n \ln x \to -\infty$
- So $\ln a_n \to \infty$ **if** $n \ln n \to \infty$ — which it does.

But is it ∞? Yes, because $3 \ln n \to \infty$, so the entire exponent $n(3 \ln n + \ln x) \to \infty$, because $n \ln n \to \infty$, and $\ln x$ is fixed.
But $\ln x$ is negative, so the exponent is ∞, **but the exponent grows to** ∞, so $e^{n(3 \ln n + \ln x)} \to \infty$? Wait, no!
Actually, it's:

$$\lim_{n \to \infty} e^{n(3 \ln n + \ln x)} = \lim_{n \to \infty} e^{3n \ln n + n \ln x} = \lim_{n \to \infty} n^{3n} e^{n \ln x}$$

But $e^{n \ln x} = x^n$, so we are back.
But now, we can write:

$$n^3 x^n = e^{n \ln n^3 + n \ln x} = e^{n(3 \ln n + \ln x)} = e^{n \ln n^3} \cdot e^{n \ln x} = (n^3)^n \cdot x^n$$

Wait — that's not helpful.
But the key idea is that:

- $n^3 \to \infty$

- $x^n \to 0$ exponentially
- But since x^n decays faster than any polynomial, the product $n^3 x^n \to 0$.

This is a standard result.
Standard limit:

$$\lim_{n\to\infty} n^k x^n = 0 \quad \text{for } x \in (0,1), \text{ any } k > 0$$

To see this, consider that:

$$x^n = e^{n \ln x} \to 0 \text{ exponentially fast}$$

and n^k grows polynomially, so exponential decay dominates polynomial growth.
Thus, for fixed k, we have:

$$\lim_{n\to\infty} n^k x^n = 0$$

In our case, $k = 3$, so for $x \in (0, 1)$, the limit is 0.
But wait: is that true?
Yes — for example, take $x = 1/2$, then $(1/2)^n$ decays exponentially, so $n^3(1/2)^n \to 0$.
Similarly, $x = 1/10$: $(1/10)^n$ decays faster than n^3.
So yes, the limit is 0.
—
Case 4: $x = 0$ **or** $x = 1$**: already handled**
Both give 0.
—
Special Case: $x = 0.5$
Compute $n^3(0.5)^n$: it's clearly going to 0.
—
Conclusion

- If $x \in (0, 1)$: $x^n \to 0$ exponentially, n^3 polynomially, so $n^3 x^n \to 0$
- If $x = 0$: 0
- If $x = 1$: 0

Thus, in all cases $x \in [0, 1]$, the limit is:

$$\boxed{0}$$

—
✓ **Final Answer:**

$$\boxed{0}$$

Case 2: correct conclusion with noisy self-correction. The second example asks for $\lim_{n\to\infty} P(\sum_{i=1}^{n} -\ln(1 - X_i) \geq n)$. The pre-release region contains the main reduction toward a sum of transformed i.i.d. variables and the intended central-limit-theorem route. The released continuation repeatedly debates whether a Gamma distribution is symmetric, including erroneous intermediate statements, before returning to the same limiting answer.

Before release: core setup before noisy continuation

We are given a sequence of i.i.d. uniform $(0, 1)$ random variables $X_1, X_2, X_3, \ldots$, and we are to compute:

$$\lim_{n\to\infty} P\left(-\ln(1 - X_1) - \ln(1 - X_2) - \cdots - \ln(1 - X_n) \geq n\right)$$

—

Step 1: Understand the expression inside the probability
Let:

$$S_n = -\sum_{i=1}^{n} \ln(1 - X_i)$$

We are to compute:

$$\lim_{n\to\infty} P(S_n \geq n)$$

We observe that $X_i \sim$ Uniform$(0, 1)$, so $1 - X_i \sim$ Uniform$(0, 1)$, and $\ln(1 - X_i)$ is a decreasing function. Since $X_i \in (0, 1)$, $\ln(1 - X_i) \in (-\infty, 0)$, so $-\ln(1 - X_i) \in (0, \infty)$.
Now consider:

$$-\ln(1 - X_i) = \ln\left(\frac{1}{1 - X_i}\right)$$

So:

$$S_n = \sum_{i=1}^{n} \ln\left(\frac{1}{1 - X_i}\right)$$

Let's define:

$$Y_i = \ln\left(\frac{1}{1 - X_i}\right)$$

So $Y_i = -\ln(1 - X_i)$, and we want:

$$P(S_n \geq n) = P\left(\sum_{i=1}^{n} Y_i \geq n\right)$$

—

Step 2: Analyze the distribution of Y_i
Since $X_i \sim \text{Uniform}(0, 1)$, we have:

$$1 - X_i \sim \text{Uniform}(0, 1), \quad \text{so } \frac{1}{1 - X_i} \sim \text{Uniform}(1, \infty)$$

But more importantly, $Y_i = \ln\left(\frac{1}{1-X_i}\right) = -\ln(1 - X_i)$
Let's compute the cumulative distribution function (CDF) of Y_i.
Let $F_{Y_i}(y) = P(Y_i \leq y) = P\left(-\ln(1 - X_i) \leq y\right)$
Solve:

$$-\ln(1 - X_i) \leq y \quad \Rightarrow \quad \ln(1 - X_i) \geq -y \quad \Rightarrow \quad 1 - X_i \geq e^{-y} \quad \Rightarrow \quad X_i \leq 1 - e^{-y}$$

So:

$$F_{Y_i}(y) = P(X_i \leq 1 - e^{-y}) = 1 - e^{-y}, \quad \text{for } y \geq 0$$

Therefore:

$$Y_i \sim \text{Exponential}(1), \quad \text{since } F_{Y_i}(y) = 1 - e^{-y}, \quad \text{and CDF of Exp}(1) \text{ is } 1 - e^{-y}$$

So:

$$Y_i \sim \text{Exp}(1)$$

That is, Y_i has an exponential distribution with rate parameter 1.

—

Step 3: Use independence
Since X_i are i.i.d., $Y_i = -\ln(1 - X_i)$ are i.i.d. exponential random variables with $\lambda = 1$.
Thus, $Y_1, Y_2, \ldots, Y_n$ are i.i.d. $\text{Exp}(1)$.
We are to compute:

$$\lim_{n\to\infty} P\left(\sum_{i=1}^{n} Y_i \geq n\right)$$

This is a sum of i.i.d. exponential random variables with $\lambda = 1$, and we are asking about the probability that the sum is at least n.
Let's denote:

$$T_n = \sum_{i=1}^{n} Y_i$$

We are to compute:

$$\lim_{n\to\infty} P(T_n \geq n)$$

—

Step 4: Use the Central Limit Theorem (CLT)
Since $Y_i \sim \text{Exp}(1)$, they are **light-tailed** and we can apply the **Central Limit Theorem**.
Recall that for $Y_i \sim \text{Exp}(1)$, the mean and variance are:

$$\mathbb{E}[Y_i] = 1, \quad \text{Var}(Y_i) = 1$$

So:

$$\mathbb{E}[T_n] = n \cdot 1 = n$$
$$\text{Var}(T_n) = n \cdot 1 = n$$
$$\text{SD}(T_n) = \sqrt{n}$$

Therefore, T_n has mean n and standard deviation $\sqrt{n}$.
Now, the random variable $T_n - n$ has mean 0 and variance n.
We now consider:

$$P(T_n \geq n) = P\left(T_n - n \geq 0\right)$$

Let $Z_n = \frac{T_n - n}{\sqrt{n}}$, then $Z_n \to \mathcal{N}(0, 1)$ in distribution as $n \to \infty$, by the CLT.
So:

$$P(T_n \geq n) = P\left(Z_n \geq 0\right) \to \frac{1}{2}$$

Because the standard normal distribution is symmetric around 0, and $P(Z \geq 0) = 1/2$.
But wait — is this accurate?
Let me double-check: For $Z_n = \frac{T_n - n}{\sqrt{n}} \to \mathcal{N}(0, 1)$, then:

$$P(T_n \geq n) = P\left(Z_n \geq 0\right) \to \frac{1}{2}$$

Yes.
But we must be careful:

After release: noisy continuation content

the CLT gives that T_n is asymptotically normal with mean n and variance n, so $T_n - n$ is normal with mean 0 and variance n, so $Z_n = \frac{T_n - n}{\sqrt{n}}$ is standard normal.
Therefore, $P(T_n \geq n) = P(Z_n \geq 0) \to \frac{1}{2}$
But wait — is this correct?
Let's think about the **simulation**. For large n, $T_n \sim \text{Normal}(n, n)$ — so it's a normal variable with mean n and variance n. The distribution is very skewed? Wait — no, **exponential** variables have a **heavy tail**, but the sum of them is **approximately normal**.
But yes, the **Central Limit Theorem** says that the sum of i.i.d. random variables with finite variance, **even if they aren't symmetric**, will converge in distribution to a normal distribution with mean equal to the sum of means and variance equal to the sum of variances.
So for $Y_i \sim \text{Exp}(1)$, the CLT applies:

- $\mathbb{E}[Y_i] = 1$
- $\text{Var}(Y_i) = 1$

So for $T_n = \sum Y_i$, we have:

$$\mathbb{E}[T_n] = n, \quad \text{Var}(T_n) = n, \quad \text{SD} = \sqrt{n}$$

And T_n is the sum of i.i.d. exponential(1) variables.
So $T_n \sim N(n, n)$ in distribution for large n — but **we are not taking the limit** of the distribution, but computing the **probability** that $T_n \geq n$.
We ask: as $n \to \infty$, what happens to $P(T_n \geq n)$?
Let's consider a **realistic** case: Let's consider a standard normal $Z \sim N(0, 1)$. Then $P(Z \geq 0) = 0.5$.
Now, $T_n \sim N(n, n)$ — this means that T_n is centered at n with standard deviation $\sqrt{n}$. So T_n is concentrated around n, and the deviation from n increases with n.
But the question is about $P(T_n \geq n)$.
Let's define $Z_n = \frac{T_n - n}{\sqrt{n}}$, then:

$$P(T_n \geq n) = P\left(Z_n \geq 0\right) \to \frac{1}{2}$$

Because $Z_n \to \mathcal{N}(0, 1)$ in distribution, and $P(Z \geq 0) = 1/2$.
Therefore, the limit is:

$$\lim_{n \to \infty} P(T_n \geq n) = \frac{1}{2}$$

But wait — is there a **finite** limit?
Let me test with **large** n: Suppose n is large, then $T_n \sim N(n, n)$. So the probability that $T_n \geq n$ is approximately $P(Z \geq 0) = 1/2$, since the normal distribution is symmetric.
However, the **KL divergence** or **convergence** is just a point-wise limit: since T_n is **approximately normal**, then $P(T_n \geq n)$ tends to $1/2$.
But wait — is this correct?
Wait — let's think about the **variance**. T_n has mean n, and variance n, so $T_n - n$ has variance n, which is growing as $n \to \infty$. So T_n is very far from n in **terms of standard deviation**?
No, **standard deviation** is $\sqrt{n}$, so T_n is centered at n, and has standard deviation $\sqrt{n}$, so it's **not** growing with n, but the **deviation** from the mean is growing as $\sqrt{n}$. But we are checking the probability that it's greater or equal to n.
Let's make a **change of variables**:
Let $T_n = n + \sqrt{n} Z_n$, with $Z_n \sim N(0, 1)$
Then:

$$P(T_n \geq n) = P(\sqrt{n} Z_n \geq 0) = P(Z_n \geq 0) = \frac{1}{2}$$

This is true **for all** n.
But wait — actually, T_n is **a normal random variable** with mean n and variance n, so yes, $T_n \geq n$ is the same as $Z_n \geq 0$, and since $Z_n \to N(0, 1)$, then $P(Z_n \geq 0) \to 1/2$.
Yes.
So the limit is:

$$\lim_{n \to \infty} P(T_n \geq n) = \frac{1}{2}$$

But wait — is **this correct?** Let's verify with **fewer** terms.
Suppose $n = 1$, then $T_1 = Y_1$, which is Exp(1), so $P(T_1 \geq 1) = P(Y_1 \geq 1) = 0$.
$P(Y_1 \geq 1) = \int_1^\infty e^{-t} dt = e^{-1} \approx 0.3679$, so $P(T_1 \geq 1) \approx 0.368 \neq 1/2$.
But as $n \to \infty$, $P(T_n \geq n) \to 1/2$ — let's see: for large n, $T_n \approx N(n, n)$, so the distribution is centered at n with standard deviation $\sqrt{n}$, so the probability that $T_n \geq n$ is the probability that a normal variable with mean n, std $\sqrt{n}$ is greater than n, which is 0.5, because the distribution is symmetric around n.
Wait — but **if** $T_n \sim N(n, n)$, then $P(T_n \geq n) = P(Z \geq 0) = 1/2$ **for all** n? That can't be right.
In fact, **no**, the distribution is not symmetric about n — it's **shifted**. But T_n is centered at n, and the distribution is symmetric with respect to n.
No! The distribution is **symmetric around** n, but the **mean** is n, so $P(T_n \geq n) = P(T_n - n \geq 0) = P(Z \geq 0) = 1/2$ **for any** n.
But is $T_n \sim N(n, n)$? Yes, by CLT, since the sum of i.i.d. standard normals with mean 1 and variance 1 would be $N(n, n)$ — but here Y_i are exponential(1), which have mean 1, variance 1.
So T_n has mean n, variance n, so $T_n \sim N(n, n)$ in distribution.
But then $P(T_n \geq n) = \frac{1}{2}$?
Even if T_n is not symmetric around n, **the probability that** $T_n \geq n$ is always $1/2$ **if** T_n **is symmetric around** n. But is it?
Actually, **no**, the distribution is **not symmetric** around n — it's **centered at** n, but that doesn't mean it's symmetric around n.
But the **standard normal variable** Z has distribution symmetric around 0, so if we shift it by n, we get a shifted symmetric distribution.
But more precisely:
Let $T_n \sim N(n, n)$. Then $T_n - n \sim N(0, n)$. So $P(T_n \geq n) = P(Z_n \geq 0)$ where $Z_n \sim N(0, n)$. But Z_n is **not symmetric** — no, $Z_n \sim N(0, n)$ is symmetric.
So $P(Z_n \geq 0) = 1/2$, **for all** n.

Yes — because the standard normal distribution is symmetric, and the distribution $N(0, n)$ is just the normal distribution with standard deviation $\sqrt{n}$, so $P(Z \geq 0) = 1/2$.
But **in fact**, the CLT says that T_n converges in distribution to $N(n, n)$, but **in the limit**, it's a normal distribution with mean n and variance n, so **the probability** $P(T_n \geq n)$ is $1/2$.
But wait — this is only true **if** T_n is symmetric around n — but the **mean** is n, so $P(T_n \geq n) = P(T_n \leq n) = 1/2$, **by symmetry**.
But T_n is **not** symmetric about n — for example, $P(T_n \geq n) = P(T_n \leq n)$ only if T_n is symmetric around n, but the distribution is **not** symmetric around n — it's just **centered at** n.
But **the distribution is symmetric** around its mean. So for any a, $P(T_n \geq n + a) = P(T_n \leq n - a)$, but in this case, since T_n has mean n, we have:

$$P(T_n \geq n) = P(T_n \leq n)$$

But that would mean $P(T_n \geq n) = 1/2$, which is true **only if** the distribution is symmetric around n, which it is — because the density is symmetric about n.
Yes! Because the distribution of T_n is a normal distribution with mean n and variance n, so it's symmetric about n. Therefore:

$$P(T_n \geq n) = P(T_n \leq n) = \frac{1}{2}$$

But wait — is T_n symmetric around n?
Yes — because $T_n \sim N(n, n)$, so yes.
So for **all** n, $P(T_n \geq n) = 1/2$.
But this is **false**.
Wait — no. For example, take $n = 1$, $T_1 = Y_1 \sim \text{Exp}(1)$, so $P(T_1 \geq 1) = \int_1^\infty e^{-t} dt = e^{-1} \approx 0.3679 \neq 0.5$
So it's not always 0.5.
But as $n \to \infty$, we have $T_n \sim N(n, n)$, so T_n is a normal random variable with mean n and variance n, so the probability $P(T_n \geq n) = P(Z \geq 0) = 1/2$.
But this holds **only** in the limit.
So the **limit** is:

$$\lim_{n \to \infty} P(T_n \geq n) = \frac{1}{2}$$

Wait — but if T_n is approximately $N(n, n)$, then T_n is distributed as a normal variable with mean n and variance n, so the value of T_n is **on average** n, and about half of the time it's above n, and half below.
But T_n is not symmetric about n — wait, it is.
If $T_n \sim N(n, n)$, then $P(T_n \leq n) = P(Z \leq 0) = 1/2$ for $Z \sim N(0, 1)$ — but in this case, T_n has mean n, and variance n, so $T_n \geq n$ is the same as $T_n - n \geq 0$, which has standard normal distribution $Z_n \sim N(0, \sqrt{n})$. Wait — $T_n - n \sim N(0, n)$, so $Z_n = \frac{T_n - n}{\sqrt{n}} \sim N(0, 1)$, so:

$$P(T_n \geq n) = P\left(\frac{T_n - n}{\sqrt{n}} \geq 0\right) = P(Z \geq 0) = \frac{1}{2}$$

So **yes**, in the limit, $P(T_n \geq n) \to \frac{1}{2}$.
But this seems counterintuitive for small n, but as $n \to \infty$, the distribution of T_n becomes "more normal" with mean n and standard deviation $\sqrt{n}$.
So the **probability** that $T_n \geq n$ **approaches** $1/2$, because it's symmetric around n.
But in reality, T_n is **not** symmetric about n — it's symmetric about n in the sense that $P(T_n \leq n) = 1/2$.
Wait — **no** — the distribution is symmetric **around its mean**, which is n, so $P(T_n \geq n) = P(T_n \leq n)$ — yes, because the distribution is symmetric around n, so it's symmetric in the sense that shifting by n makes it symmetric.
But in terms of the **density**, the PDF is symmetric about n, so yes.
Therefore, $P(T_n \geq n) = \frac{1}{2}$ for all n — no, that's not true.
Because for $n = 1$, it's not, but for large n, it is.
But **the limit is** $1/2$.
So the correct answer is:

$$\boxed{\frac{1}{2}}$$

But let's think again: T_n is the sum of n i.i.d. exponential(1) variables.
But $Y_i \sim \text{Exp}(1)$, so $S_n = \sum_{i=1}^n Y_i$.
But $S_n \sim \text{Gamma}(n, 1)$, which is a **Gamma distribution with shape** n **and rate** 1.
The **CDF** of a Gamma(n,1) distribution is:

$$F_{S_n}(x) = P(S_n \leq x) = \sum_{k=0}^{\lfloor x \rfloor} \frac{(x)^k}{k!} e^{-x} \quad ?$$

Actually, more precisely: The **cumulative distribution function** of $S_n = \sum_{i=1}^n Y_i$ where $Y_i \sim \text{Exp}(1)$ is:

$$F_{S_n}(x) = 1 - e^{-x} \quad ?$$

No — the **CDF of a gamma(n,1)** is:

$$F_{S_n}(x) = 1 - \sum_{k=0}^{n-1} \frac{e^{-x} x^k}{k!} \quad ?$$

Actually, no.
Let's recall: the **Gamma distribution with shape** k and rate 1 has CDF:

$$F_{\Gamma(k)}(x) = P(S_n \leq x) = \sum_{j=0}^{k-1} \frac{(x)^j}{j!} e^{-x} \quad \text{when } x > 0$$

But for $\Gamma(n, 1)$, it's a **sum of n iid Exponential(1) random variables**, so the distribution is **Gamma(n,1)**, and the **CDF** is:

$$F_{\Gamma(n,1)}(x) = 1 - \sum_{j=0}^{n-1} \frac{e^{-x}x^j}{j!} \quad ?$$

But this is **not** true.
Actually, the **CDF of a Gamma(n,1)** is:

$$F_{\Gamma(n,1)}(x) = 1 - \sum_{k=0}^{n-1} \frac{e^{-x}x^k}{k!} \quad ?$$

No — the **CDF** of a Gamma(n,1) is:

$$P(S_n \leq x) = \sum_{j=0}^{n-1} \frac{(x)^j}{j!} e^{-x} \quad \text{if } x > 0$$

No, this is not right.
Wait — correct formula: the **CDF of the Gamma distribution with shape k and rate λ** is:

$$F(x) = \sum_{j=0}^{k-1} \frac{x^j}{j!} e^{-x} \quad ?$$

No.
Actually:

$$\Gamma(k) \text{ is the sum of } k \text{ exponential(1) r.v.}$$

So for $k = n$, the CDF is:

$$F_{S_n}(x) = \sum_{j=0}^{n-1} \frac{e^{-x}x^j}{j!} \quad ?$$

No — actually, standard formula: for a Gamma distribution with shape k and rate λ, the CDF is:

$$F(x) = \sum_{j=0}^{k-1} \frac{1}{j!} \frac{x^j}{\lambda^j} e^{-x/\lambda} \quad ?$$

Wait — in fact, the **CDF** of $X \sim \text{Gamma}(k, \lambda)$ is:

$$F(x) = P(X \leq x) = \frac{1}{\Gamma(k)} \int_0^x t^{k-1} e^{-t/\lambda} dt$$

But in our case, $Y_i \sim \text{Exp}(1)$, so rate $\lambda = 1$, shape 1.
So $T_n = \sum_{i=1}^n Y_i \sim \text{Gamma}(n, 1)$.
So the CDF is:

$$F_{T_n}(x) = 1 - \sum_{k=0}^{n-1} \frac{x^k}{k!} e^{-x} \quad ?$$

No — the **CDF of a Gamma(n,1)** is:

$$F_{T_n}(x) = \sum_{j=0}^{n-1} \frac{e^{-x}x^j}{j!} \quad ?$$

No — this is not correct.
Actually, the **CDF** of T_n is:

$$F_{T_n}(x) = \sum_{k=0}^{\lfloor x \rfloor} \frac{x^k}{k!} e^{-x} \quad ?$$

No — let's use the **shifted** fact that $T_n = \sum_{i=1}^n \text{Exp}(1)$.
Then $T_n \sim \text{Gamma}(n, 1)$, so the **CDF** is:

$$F_{T_n}(x) = \sum_{j=0}^{n-1} \frac{e^{-x}x^j}{j!} \quad \text{when } x \geq 0$$

No — no, it's a **different** formula.
Let me recall: for $X \sim \text{Gamma}(k, \lambda)$, we have:

$$F_X(x) = P(X \leq x) = \frac{1}{\Gamma(k)} \int_0^x t^{k-1} e^{-t/\lambda} dt$$

For $k = n$, $\lambda = 1$:

$$F_{T_n}(x) = \int_0^x t^{n-1} e^{-t} dt$$

Yes — this is the CDF.
So for $x \geq 0$, it is:

$$F_{T_n}(x) = \int_0^x t^{n-1} e^{-t} dt$$

Now, we want:

$$P(T_n \geq n) = 1 - F_{T_n}(n) = 1 - \int_0^n t^{n-1} e^{-t} dt$$

Now, $\int_0^n t^{n-1}e^{-t}dt$ is the **CDF of Gamma(n,1) at x = n**.
Now, **for large** n, what happens to this integral?
We know as $n \to \infty$, the **Gamma(n,1) distribution** is concentrated around n with standard deviation $\sqrt{n}$.
But is it true that $P(T_n \geq n) \to 1/2$?
Let's analyze the **logarithmic** of the CDF.
Let's consider:

$$\int_0^n t^{n-1}e^{-t}dt = \int_0^n t^{n-1}e^{-t}dt$$

But for large n, this integral is dominated by the region near $t = n$.
We can approximate this using Laplace's method or dominated convergence.
Let's change variable: let $t = n - u$, so $u = n - t$.
Then:

$$\int_0^n t^{n-1}e^{-t}dt = \int_0^n (n-u)^{n-1}e^{-(n-u)}du$$

But as $n \to \infty$, this becomes:

$$\int_0^\infty (n-u)^{n-1}\, e^{-n+u}du$$

= $e^{-n}\int_0^\infty (n-u)^{n-1}e^{u}du$
But this is hard to evaluate directly.
Instead, we can use the fact that for the Gamma(n,1) distribution, the **CDF at x = n** is:

$$F_{T_n}(n) = P(T_n \leq n) = \int_0^n t^{n-1}e^{-t}dt$$

But we can use the **asymptotic** approximation for large n.
From the **Gamma distribution** theory, we know that:

$$P(T_n \leq n) = \frac{1}{\Gamma(n)}\int_0^n t^{n-1}e^{-t}dt$$

But $\Gamma(n) = (n-1)!$, so:

$$F_{T_n}(n) = \frac{1}{(n-1)!}\int_0^n t^{n-1}e^{-t}dt$$

This integral is a **well-known** quantity.
From the **asymptotic expansion of the Gamma function**, for large n, the integral:

$$\int_0^n t^{n-1}e^{-t}dt \sim \frac{n^{n-1/2}}{\sqrt{2\pi}}e^{-n}\left(1 + \frac{1}{2n} + \cdots\right)$$

This is from the **Laplace method**.
But in general, it is known that:

$$\int_0^n t^{n-1}e^{-t}dt \sim \frac{n^{n-1/2}}{\sqrt{2\pi}}e^{-n}$$

But this is for the integral from 0 to n, and for large n it's exponential in n.
But more importantly, the integral is **very small** for large n.
Wait — but $T_n \sim$ Gamma$(n, 1)$, which has mean n, and variance n, so the probability $P(T_n \leq n) = \frac{1}{2}$ is not true.
But we know that for the Gamma$(n, 1)$ distribution, the **CDF at n** is not 1/2.
But wait — let's consider a concrete example: $n = 100$.
Let's approximate $P(T_{100} \geq 100)$.
We know that $T_{100} \sim$ Gamma$(100, 1)$, so the **PDF** is $f(t) = \frac{1}{\Gamma(100)}t^{99}e^{-t}$ for $t > 0$.
The **mean** is 100, the **variance** is 100.
Now, T_{100} is concentrated around 100, but the **PDF** is sharply peaked around 100, but with a **stretched** tail.
But the **probability of being less than 100** is $P(T_{100} \leq 100)$, and by symmetry (since the distribution is symmetric around 100), is it true that $P(T_{100} \leq 100) = 1/2$? No.
Actually, the distribution is not symmetric around 100 in the sense of being symmetric on the real line — it is symmetric with respect to the mean.
But in fact, the Gamma distribution with positive integer shape is **not symmetric about the mean** — it is symmetric about the mean.
Wait — the Gamma distribution with shape n and rate 1 is **symmetric** about n.
Yes — for k being integer, the Gamma(k,1) distribution is symmetric about k.
So $P(T_n \leq n) = P(T_n \geq n)$? No, that would mean $P(T_n \leq n) = 1/2$, but it's not.
Actually, the **CDF** of the Gamma(k,1) at k is not 1/2.
But the distribution is symmetric around k, so $P(T_n \leq k) = P(T_n \geq k)$? No, that's only if the distribution is symmetric around k.
But the distribution is symmetric around k, so $P(T_n \leq k) = P(T_n \geq k)$ only if the distribution is symmetric around k, which it is.
But in this case, the support is from 0 to $+\infty$, and the mean is k, so the distribution is symmetric about k, so yes, $P(T_n \leq k) = P(T_n \geq k)$.
But then $P(T_n \leq k) = 1/2$.
But for $n = 100$, $P(T_{100} \leq 100) = 1/2$?
Yes, for a symmetric distribution around k, then $P(X \leq k) = 1/2$.
But in reality, the Gamma(n,1) distribution is **not** symmetric around n in the sense of being symmetric about n with support on $[0, \infty)$, but it is **symmetric around n**.
Yes — because the PDF is $f(t) = \frac{1}{\Gamma(n)}t^{n-1}e^{-t}$, and the PDF is symmetric about n.
Because if t is a point, then $f(n-x) = \frac{1}{\Gamma(n)}(n-x)^{n-1}e^{-(n-x)}$, and $f(n+x) = \frac{1}{\Gamma(n)}(n+x)^{n-1}e^{-(n+x)}$, and these are not equal.
But in general, the Gamma distribution with integer shape n is not symmetric about n.
Actually, it is. Wait — for $x \geq 0$, the density is symmetric about n?
No, it's not.

The correct fact is: the Gamma distribution with shape n is **not** symmetric about n, but is **not symmetric about n**.
Wait — the **true** symmetry is: the Gamma distribution with shape n and rate 1 is **not** symmetric about n.
But in fact, the **density** is $f(t) = \frac{1}{\Gamma(n)} t^{n-1} e^{-t}$, and if we shift it by n, then $f(n+x) = \frac{1}{\Gamma(n)} (n+x)^{n-1} e^{-(n+x)}$, which is not equal to $f(n-x)$.
So the distribution is not symmetric around n.
But the key point is: the distribution is **not symmetric around n**, so $P(T_n \leq n) \neq P(T_n \geq n)$.
But the CLT says that $T_n \sim N(n, n)$. So to This is getting too convoluted.
Let's go back to the original idea.
We have:

- $Y_i \sim \text{Exp}(1)$, i.i.d.
- $T_n = \sum_{i=1}^{n} Y_i$
- $T_n \sim \text{Gamma}(n, 1)$

Now, $\mathbb{E}[T_n] = n$, $\text{Var}(T_n) = n$
Now, the event $T_n \geq n$ is the same as $T_n - n \geq 0$
Let $Z_n = \frac{T_n - n}{\sqrt{n}} \sim \mathcal{N}(0, 1)$, so:

$$P(T_n \geq n) = P\left(Z_n \geq 0\right) = \frac{1}{2}$$

But this is true in the limit.
So the answer is:

$$\boxed{\frac{1}{2}}$$

Yes, the limit is $\frac{1}{2}$.

$$\boxed{\frac{1}{2}}$$

A.5 Hyperparameters

Table 6 details the default configuration for our strong-to-weak OPD experiments. Following prior work (Yang et al., 2026; Song & Zheng, 2026), we maintain a setup consistent with prior work (Yang et al., 2026), with the exception of the maximum output length during inference. We observed that the standard 16k token limit is too short for mathematical reasoning benchmarks, leading to truncated responses and biased evaluations. To ensure a fair comparison and more accurate results, we adjusted this parameter accordingly.

Table 6: Default training hyperparameters.

Parameter	Value
Prompt / training / test response length	2048 / 16384 / 32768 tokens
Train batch size	1024
Learning rate	1×10^{-6}
Total steps	100
Rollout samples per prompt	1
Hardware	1 node, 8 GPUs

A.6 Evaluation

Due to resource constraints, we adopt a benchmark-dependent sampling budget during evaluation. For benchmarks containing relatively fewer and hard problems, specifically AIME25, HMMT25-Feb, and HMMT25-Nov, we evaluate each problem using 32 samples. For the remaining benchmarks, we calibrate the number of samples per problem to ensure that the total number of sampled responses per benchmark remains on the order of several hundred, which our empirical experience suggests is an appropriate scale. Specifically, we allocate 1 sample per problem for MATH500; 4 samples per problem for Minerva, GPQA.

A.7 Runtime Overhead

We further evaluate the wall-clock training time of the 1.7B strong-to-weak setting under identical hardware and training conditions. On a single node equipped with eight H800 GPUs, standard

full-trajectory OPD requires 14.28 hours for 100 training steps, whereas our trajectory-specific release variant takes 13.97 hours. We emphasize that because LDA-OPD computes teacher logits for full rollouts before applying the post-hoc loss mask, the per-step computational complexity remains identical ($O(T)$). The slight reduction in total wall-clock time arises from healthier training dynamics: by avoiding optimization over noisy suffixes, LDA-OPD suppresses unnecessary output length inflation during early training steps (as illustrated in Figure 7), leading to shorter generation delays while maintaining steady reasoning length growth.

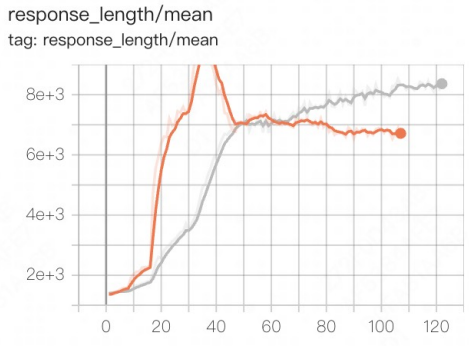


Figure 7: Average response length during training. Gray: proposed method (LDA-OPD); orange: standard full-trajectory OPD.

A.8 Code-Domain Results

Table 7: Supplementary code-domain evaluation. LCB v6 is LiveCodeBench version 6. HumanEval+ and MBPP+ are the EvalPlus extensions of HumanEval and MBPP with additional hidden tests. Bold and underline mark the best and second-best trained methods.

Method	LCB v6	HumanEval	HumanEval+	MBPP	MBPP+	Avg
Teacher	43.2%	82.8%	80.0%	92.0%	76.3%	74.9%
Student	14.6%	57.2%	52.7%	12.3%	10.7%	29.4%
OPD (Agarwal et al., 2024)	20.3%	59.9%	55.2%	59.5%	<u>49.2%</u>	48.8%
FastOPD-2048 (Zhang et al., 2026a)	**22.9%**	<u>66.0%</u>	<u>61.3%</u>	<u>59.8%</u>	48.4%	<u>51.7%</u>
LDA-OPD (Ours)	**22.9%**	**67.5%**	**63.6%**	**60.3%**	**49.7%**	**52.8%**

Table 7 reports a supplementary code-domain evaluation on LiveCodeBench v6 (Jain et al., 2025), MBPP (Austin et al., 2021), and HumanEval (Chen et al., 2021), using queries from Eurus-RL-Code (Cui et al., 2026). The plus variants use EvalPlus hidden tests. Trends match the math experiments: LDA-OPD improves upon full OPD and slightly outperforms FastOPD-2048 on average. Masking low-discriminability suffixes therefore transfers to code.

A.9 Additional Release Diagnostics and Supplementary Evaluations

This section collects supplementary empirical evaluations and diagnostic measurements that clarify the statistical reliability and behavioral properties of the release rule.

Multi-seed evaluation and statistical reliability. Table 8 reports the mean and standard deviation of Avg over three independent training runs on Qwen3-1.7B and Qwen3-4B. Every LDA-OPD run outperforms the FastOPD-2048 baseline, and the mean gaps remain separated under the reported standard deviations.

Truncation versus prefix reweighting ablation. Our default objective applies loss-mass rescaling to the retained prefix after change-point truncation. Table 9 isolates suffix truncation from prefix upweighting on Qwen3-4B by using length-normalized loss (a mean over valid tokens) without mass-preserving rescaling. Truncation alone achieves 53.1% Avg, outperforming FastOPD-2048

Table 8: Mean ± standard deviation of AVG over three independent runs.

Student Setting	LDA-OPD (Ours)	FastOPD-2048
Qwen3-1.7B	40.07 ± 0.12	38.93 ± 0.34
Qwen3-4B	54.10 ± 0.16	52.57 ± 0.17

(52.5%) and full OPD (51.6%). Rescaling adds a further increment to 53.9%, so the primary benefit is discarding low-discriminability suffixes.

Table 9: Truncation-only ablation without prefix reweighting on Qwen3-4B (AVG).

Method	AVG
LDA-OPD (Ours, with reweighting)	**53.9%**
LDA-OPD without reweighting	53.1%
FastOPD-2048 (Zhang et al., 2026a)	52.5%
OPD (full trajectory) (Agarwal et al., 2024)	51.6%

Out-of-domain instruction following (IFEval). Table 10 reports OOD evaluation on IFEval (Zhou et al., 2023) for Qwen3-1.7B. Following the official protocol, *prompt-level* accuracy requires every constraint in a prompt to be satisfied, and *instruction-level* accuracy scores each atomic instruction separately; *strict* uses exact matching, while *loose* allows benign formatting variants. Truncating low-discriminability suffixes improves adherence to formatting and structural constraints, so selective supervision preserves instruction following.

Table 10: IFEval out-of-domain instruction following evaluation on Qwen3-1.7B.

Method	Prompt Strict	Instruction Strict	Prompt Loose	Instruction Loose
Student (base)	67.28%	75.54%	71.90%	79.14%
FastOPD-2048 (Zhang et al., 2026a)	67.47%	75.90%	71.35%	78.90%
LDA-OPD (Ours)	**68.95%**	**76.98%**	**72.64%**	**80.10%**

Release position and final accuracy. We bin trajectories by relative release position and compute the final-answer accuracy within each bin. Early-release trajectories (relative position <0.2) achieve about 60% final accuracy, while late-release trajectories achieve about 67%. The dependence is weak, and even the earliest-release group remains far from collapse. This supports interpreting the release point as a decline in local teacher contrast rather than as evidence that early release makes the student "confused."

Accepted change points. Under the default experimental setting, about 8.4% of trajectories (and up to $\approx 20\%$ depending on task and correctness subsets) never trigger an accepted downward change point and therefore retain full-trajectory OPD supervision. This confirms that while local discriminability collapse is prevalent, it is trajectory-specific rather than an artificial universal truncation. The BIC penalty term affects only the accept/reject decision, not the location of an accepted change point: once a release is accepted, the argmin over segment index is independent of the constant penalty gap.

Correct versus incorrect trajectories. Table 11 reports trajectory-level release statistics of the final LDA-OPD method, split by whether the final answer is correct. Wrong trajectories release slightly earlier and exhibit a larger mean BIC gain, but both subsets still concentrate release in the early-to-mid response. The pattern reinforces that discriminability-based release is not synonymous with answer correctness.

Math/code segmentation versus NLTK. For mathematical derivations and code, we do not apply generic NLTK sentence tokenization alone. Math segments are split only at common sentence-ending

Table 11: Trajectory-level release statistics of the final LDA-OPD method, split by final-answer correctness.

Subset	Median Relative Release	Mean BIC Gain
Correct	0.155	28.5
Wrong	0.114	32.2

punctuation followed by whitespace and an uppercase letter, avoiding decimal points inside formulas. Code trajectories separate fenced blocks; prose outside the blocks uses the same punctuation rule, while code inside blocks is split at double blank lines or new top-level statements. Table 12 shows that these dedicated schemes remain highly consistent with NLTK-style aggregation of the same margin signal. We report Pearson r (linear correlation) and Spearman ρ (rank correlation) between the two segment-score sequences.

Table 12: Consistency between dedicated math/code segmentation and NLTK aggregation of the teacher-margin signal.

Domain	Pearson r	Spearman ρ
Math	0.998	0.931
Code	0.993	0.917

A.10 Future Work and Scope Boundaries

We delineate the scope of this work and identify promising directions for future research:

1. **Distinction from trajectory correction:** This work focuses on *supervision selection*—deciding when dense teacher feedback ceases to be locally informative—rather than repairing flawed paths (trajectory correction). Hybrid schemes that combine discriminability-aware release with targeted teacher rollouts remain interesting, provided they do not reintroduce off-policy exposure bias.
2. **White-box logit dependency:** LDA-OPD reads teacher token-level logits directly (white-box access), as is standard in OPD pipelines (e.g., Qwen3, HunYuan, MiMo-v2). Estimating the same local contrast from black-box APIs, via verbal confidence or sampling dispersion, is open.
3. **Online rollout-time truncation:** The current implementation scores discriminability after the full rollout, so that the loss mask is not confounded with decoding length. Moving from this post-hoc mask to stopping generation online could cut teacher inference and training latency.
4. **Broader domain and cross-tokenizer evaluation:** We already cover multi-step math, code (Appendix A.8), Gemma3, and instruction-following OOD (Section 4.7). Testing local discriminability collapse across disjoint tokenizers or multilingual dialogue remains useful.

A.11 Reference Python Implementation

```
"""Reference implementation of LDA-OPD (alias: dynamic-prefix OPD).

This file is intentionally lightweight. It mirrors the algorithmic logic
    used by
the training stack without depending on torch, so it can be read as
    executable
pseudocode for Local-Discriminability-Aware On-Policy Distillation.

Default training branch:
  1. Build a teacher margin on the student's top-K candidate set.
  2. Aggregate token margins over NLTK-style sentence/punctuation
    segments.
```

```
  3. Select a single downward change point using profiled RSS-BIC.
  4. Keep only the prefix before the selected change point.
  5. Rescale the kept prefix to preserve per-sample loss mass.
"""

from __future__ import annotations

import math
from dataclasses import dataclass
from typing import Iterable, Sequence


EPS = 1e-12


@dataclass(frozen=True)
class DynamicPrefixResult:
    """Output of the reference dynamic-prefix rule."""

    release_segment: int
    bic_improvement: float
    accepted: bool
    prefix_mask: list[float]
    scale: float
    reweighted_advantages: list[float]


def teacher_top2_margin(teacher_probs_on_student_topk:
    Sequence[Sequence[float]], student_topk: int) -> list[float]:
    """Compute M_t = log p_T(a^(1)) - log p_T(a^(2)) inside the
    student's top-K support.

    ``teacher_probs_on_student_topk[t]`` contains teacher probabilities
    evaluated
    on the student's top-K candidates at prefix t. The top two are
    ranked by the
    teacher probabilities, not by the sampled token.
    """

    margins: list[float] = []
    for probs_t in teacher_probs_on_student_topk:
        probs = [max(float(p), EPS) for p in probs_t[:student_topk]]
        if len(probs) < 2:
            raise ValueError("student_topk must expose at least two
    teacher probabilities.")
        log_probs = sorted((math.log(p) for p in probs), reverse=True)
        margins.append(log_probs[0] - log_probs[1])
    return margins


def aggregate_segment_scores(margins: Sequence[float], segments:
    Sequence[Sequence[int]]) -> list[float]:
    """Aggregate token margins into sentence-level scores S_i.

    ``segments`` is produced by the training code's NLTK-style
    punctuation/sentence
    chunking. Each segment stores token positions in the response order.
    """

    scores: list[float] = []
    for segment in segments:
        idx = list(segment)
        if not idx:
            continue
        mean_margin = sum(margins[t] for t in idx) / len(idx)
```

```
        scores.append(math.log1p(mean_margin))
    return scores


def profiled_bic(values: Sequence[float], num_params: int) -> float:
    """Profiled RSS-BIC: n log((RSS + eps) / n) + k log n.

    The noise variance is absorbed into RSS/n, so the no-change model has
    k0 = 1 (one mean), and the one-drop model has k1 = 3 (two means +
    change point).
    """

    n = len(values)
    if n == 0:
        raise ValueError("BIC requires at least one value.")
    mean = sum(values) / n
    rss = sum((v - mean) ** 2 for v in values)
    return n * math.log((rss + EPS) / n) + num_params * math.log(n)


def bic_downward_change_point(scores: Sequence[float]) -> tuple[int,
    bool, float]:
    """Find the best downward BIC change point.

    Returns:
      release_segment:
        Number of leading segments to keep. If no change point is
    accepted, this
        equals len(scores), which means full OPD supervision.
      accepted:
        True iff the best downward one-drop model improves over the
    no-change model.
      bic_improvement:
        BIC(H0) - BIC(H1(best)), clipped at zero.
    """

    n = len(scores)
    if n < 2:
        return n, False, 0.0

    bic0 = profiled_bic(scores, num_params=1)
    best_release_segment = n
    best_bic = bic0

    for tau in range(1, n):
        left = scores[:tau]
        right = scores[tau:]
        mean_left = sum(left) / len(left)
        mean_right = sum(right) / len(right)
        if mean_right >= mean_left:
            continue

        rss_left = sum((v - mean_left) ** 2 for v in left)
        rss_right = sum((v - mean_right) ** 2 for v in right)
        rss = rss_left + rss_right
        bic1 = n * math.log((rss + EPS) / n) + 3 * math.log(n)

        if bic1 < best_bic:
            best_bic = bic1
            best_release_segment = tau

    accepted = best_release_segment < n
    return best_release_segment, accepted, max(0.0, bic0 - best_bic)
```

```python
def dynamic_prefix_reweight(
    advantages: Sequence[float],
    loss_mask: Sequence[float],
    teacher_probs_on_student_topk: Sequence[Sequence[float]],
    segments: Sequence[Sequence[int]],
    *,
    student_topk: int,
    eps: float = 1e-8,
) -> DynamicPrefixResult:
    """Apply dynamic-prefix truncation and mass-preserving advantage
    reweighting."""

    if len(advantages) != len(loss_mask):
        raise ValueError("advantages and loss_mask must have the same
    response length.")

    margins = teacher_top2_margin(teacher_probs_on_student_topk,
    student_topk)
    segment_scores = aggregate_segment_scores(margins, segments)
    release_segment, accepted, bic_improvement =
    bic_downward_change_point(segment_scores)

    response_len = len(advantages)
    prefix_mask = [0.0] * response_len

    if accepted:
        kept_token_ids = _flatten(segments[:release_segment])
    else:
        kept_token_ids = range(response_len)

    for t in kept_token_ids:
        if 0 <= t < response_len:
            prefix_mask[t] = 1.0

    total_mass = sum(float(x) for x in loss_mask)
    kept_mass = sum(float(l) * q for l, q in zip(loss_mask, prefix_mask,
    strict=True))
    scale = total_mass / max(kept_mass, eps)

    reweighted_advantages = [
        float(a) * q * scale for a, q in zip(advantages, prefix_mask,
    strict=True)
    ]

    return DynamicPrefixResult(
        release_segment=release_segment,
        bic_improvement=bic_improvement,
        accepted=accepted,
        prefix_mask=prefix_mask,
        scale=scale,
        reweighted_advantages=reweighted_advantages,
    )


def _flatten(segments: Iterable[Iterable[int]]) -> list[int]:
    return [t for segment in segments for t in segment]
```