LEARNING FROM HOW LARGE LANGUAGE MODELS HAVE CHANGED

Anonymous authors
Paper under double-blind review

ABSTRACT

Supervised fine-tuning (SFT) and on-policy distillation (OPD) improve language models through demonstrations and teacher feedback. However, the learner's current state alone may not reveal how these signals should be allocated for further task improvement. We introduce REWIND, which uses the model's responses to prior training to scale each token's existing loss gradient with a positive, clipped historical guide signal. The same approach supports SFT with cached earlier predictions and OPD with historical-reference scores on student rollouts. Our analysis characterizes how target-probability change and competing-token disagreement shape the actual clipped multiclass update, and gives sufficient conditions for one-step task-risk improvement. Experiments across multiple task domains and two model scales demonstrate consistent gains in both SFT and OPD.

1 INTRODUCTION

Post-training adapts a pretrained language model to downstream tasks through external supervision. Supervised fine-tuning (SFT) learns from demonstrated responses (Ouyang et al., 2022), whereas on-policy distillation (OPD) learns from teacher feedback on the student's own generations (Gu et al., 2024; Agarwal et al., 2024). In both settings, the supervision specifies how each prediction should be corrected. Training also changes the model itself: on a fixed prefix, the probability assigned to a supervised token may have risen or fallen relative to an earlier checkpoint. We ask whether this predictive change can inform how subsequent supervision is allocated.

Recent work allocates supervision across tokens according to the learner's current predictions. In SFT, token-level loss reweighting based on current target probabilities aims to improve generalization and mitigate overfitting (Wu et al., 2026; Li et al., 2026a; Sabbaghi et al., 2026). In OPD, student uncertainty and teacher–student disagreement locate positions where teacher feedback is most informative (Xu et al., 2026). These criteria primarily describe where the learner currently stands, not how it arrived there: two positions with the same current target probability may have gained or lost probability during training, yet a rule that reads only current target confidence assigns them the same weight. Comparing the current model with a frozen earlier checkpoint on the same prefix exposes this change for each token. We investigate whether it can guide token-level supervision toward better downstream performance.

To use this signal, we introduce REWIND, which conditions supervision on the learner's predictive change from a frozen earlier checkpoint. On each matched prefix, REWIND geometrically interpolates the historical and current predictions on a shared vocabulary partition. The resulting reference sets a bounded, detached historical guide signal for each original token loss. Demonstrations supply the SFT targets, teacher feedback supplies OPD corrections, and the signal sets each term's contribution to the update. Fixed demonstration prefixes allow offline reference scoring for SFT, while OPD scores the frozen checkpoint on student rollouts.

Analysis relates earlier displacement to unfinished adaptation, characterizes the clipped multiclass weights, and derives a conditional one-step risk comparison. Across two model scales, REWIND improves mathematics and code averages in SFT and OPD, with additional SFT gains on logic and instruction following. Matched ablations isolate token assignment, competing-token information, and reference timing.

Our contributions are:

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.

- **Training history as a token-level signal.** We compare the current student with a frozen earlier checkpoint on the same prefix. Tokens with identical current confidence can thereby receive different historical guide signals according to how their predictions have changed.
- **A shared rule for SFT and OPD.** A normalized geometric reference yields positive, clipped historical guide signals that allocate demonstration and teacher feedback across token positions.
- **Evidence across training settings.** We evaluate SFT and OPD on mathematics and code at two model scales, with further SFT results on logic and instruction following. Matched ablations test token-specific assignment, competing-token information, historical fusion, reference timing, and clipping.

2 Related Work

Supervised fine-tuning and token allocation. Several methods adjust the contribution of demonstrated tokens using the student's current predictions. Probability-based SFT objectives and InfoSFT use target-token confidence to shape the training loss (Wu et al., 2026; Li et al., 2026a; Sabbaghi et al., 2026), while EAFT uses predictive entropy to identify confident conflicts (Diao et al., 2026). Reference-based selection includes RHO-LOSS, which estimates reducible loss for examples with a holdout-trained model (Mindermann et al., 2022), and Rho-1, which selects pretraining tokens by excess loss against an additional trained reference (Lin et al., 2024). REWIND weights SFT and OPD supervision using predictive change between a frozen earlier learner checkpoint and the current student.

On-policy distillation. OPD trains on student-generated prefixes with teacher feedback (Gu et al., 2024; Agarwal et al., 2024). DistiLLM uses adaptive off-policy reuse of student-generated outputs to improve distillation efficiency (Ko et al., 2024). Recent methods allocate that feedback using student–teacher probability ratios, token position, trajectory quality, or uncertainty and disagreement (Lin et al., 2026; Liu et al., 2026; Li et al., 2026b; Xu et al., 2026). REWIND-OPD additionally compares the current student with an earlier frozen checkpoint to weight teacher-distillation terms.

Concurrent work Concurrent work also uses reference models to guide token supervision. W2S-OPD (Yu et al., 2026) constructs a proxy teacher by adding the logit difference between two weaker models to the student's frozen base model; AD-OPSD (Peng et al., 2026) adaptively blends the frozen base and teacher distributions to form token-level targets. REWIND compares a frozen earlier checkpoint with the evolving student on matched prefixes and uses their geometric reference to calibrate the strength of existing supervision.

3 Theory

We ask when a model's prior change can indicate unfinished task adaptation and how the implemented REWIND weight affects the next supervised update. Throughout, θ_0 is the parameter vector of the selected frozen historical checkpoint π_0. The controlled statistical model starts its trajectory at θ_0; the clipped-weight identities and one-step risk comparison hold for any fixed historical reference. Proofs and further special cases appear in Appendix B.

3.1 When can prior learning reveal a remaining correction?

Let $u = \theta - \theta_0$, with $u(0) = 0$ at the start of the modeled trajectory. In the model

$$\widehat{L}(u) = \tfrac{1}{2}\sum_j \gamma_j (u_j - \mu_j - \xi_j)^2, \qquad R_{\mathrm{quad}}(u) = \tfrac{1}{2}\sum_j \gamma_j (u_j - \mu_j)^2, \tag{1}$$

μ is the task increment, $\gamma_j > 0$, and sampling variation obeys $\mathbb{E}\xi_j = 0$ and $\mathbb{E}\xi_j^2 = v_j < \infty$. SFT gradient flow gives $u_j(\tau) = \rho_j(\mu_j + \xi_j)$ with $\rho_j = 1 - e^{-\gamma_j \tau}$ (Ali et al., 2019).

Proposition 3.1 (Prior learning and remaining task correction)**.** *Under Eq. (1),*

$$A_j(\tau) := \mathbb{E}[u_j(\tau)(\mu_j - u_j(\tau))] = \rho_j\big[(1-\rho_j)\mu_j^2 - \rho_j v_j\big]. \tag{2}$$

For $\tau > 0$ and $\mu_j \neq 0$, $A_j(\tau) > 0$ exactly when $\rho_j < \mu_j^2/(\mu_j^2 + v_j)$.

In this quadratic trajectory, prior change aligns with the remaining task correction while unfinished adaptation outweighs accumulated sampling variation. The sign condition identifies when that history is informative. The controlled continuation-risk calculation is in Appendix B.7.

3.2 How does the historical guide signal encode this change?

Fix n supervised token states and $P_i(\theta) = \mathrm{softmax}(z_i(\theta))$. Let $p_i = P_i(\theta)_{y_i}$, $q_i = P_i(\theta_0)_{y_i}$, $\Delta_i = \mathrm{logit}(p_i) - \mathrm{logit}(q_i)$, and $g_i = \nabla_\theta[-\log p_i]$. A fixed partition $\mathcal{P}_i$ contains the target singleton; write $u_{0,i}$ and $u_{\theta,i}$ for the conditional distributions on its non-target cells. Their overlap defines the competitor correction

$$D_i^{\mathcal{P}} = -\log \sum_{C \neq \{y_i\}} u_{0,i}(C)^{\lambda} u_{\theta,i}(C)^{1-\lambda} \geq 0. \tag{3}$$

Set $W_p(s) = \mathrm{clip}((1 - p + pe^{-s})^{-1}, \kappa^{-1}, \kappa)$ for $\lambda \in [0, 1]$ and $\kappa \geq 1$.

Theorem 3.1 (Exact clipped multiclass weighting)**.** *For positive cell probabilities, the detached weights satisfy*

$$w_i^{\mathcal{P}} = W_{p_i}(\lambda\Delta_i - D_i^{\mathcal{P}}), \qquad G_{\mathcal{P}} = \frac{1}{n}\sum_i w_i^{\mathcal{P}} g_i. \tag{4}$$

If $\mathcal{Q}_i$ refines $\mathcal{P}_i$ and $d_i = D_i^{\mathcal{Q}} - D_i^{\mathcal{P}} \geq 0$, then

$$e^{-d_i} w_i^{\mathcal{P}} \leq w_i^{\mathcal{Q}} \leq w_i^{\mathcal{P}}. \tag{5}$$

For $\kappa > 1$, amplification occurs exactly when $\lambda\Delta_i > D_i^{\mathcal{P}}$.

Binary pooling has $D_i = 0$. A finer partition retains changes among competing tokens, which can temper amplification even when target support has increased. The result includes active clipping and nonlinear shared parameters. Appendix B.3 further relates the update to parameter displacement and gives its local positive-semidefinite first-order correction.

3.3 When does the next update improve task risk?

Let $R(\theta)$ be a specified differentiable task risk with L-Lipschitz gradient on the update region. Define $G_{\mathrm{S}} = n^{-1}\sum_i g_i$, $\delta G = G_{\mathcal{P}} - G_{\mathrm{S}}$, and $a_i = \langle \nabla R(\theta), g_i \rangle$. The scalar a_i measures how a token's negative gradient aligns locally with risk reduction.

Theorem 3.2 (Risk comparison for the implemented update)**.** *From the same checkpoint and batch, take $\theta_{\mathrm{S}}^+ = \theta - \eta G_{\mathrm{S}}$ and $\theta_{\mathcal{P}}^+ = \theta - \eta G_{\mathcal{P}}$. Let*

$$\mathcal{A}_{\mathcal{P}} = \frac{1}{n}\sum_i (w_i^{\mathcal{P}} - 1) a_i, \qquad \mathcal{B}_{\mathcal{P}} = L\|\delta G\|(\|G_{\mathrm{S}}\| + \tfrac{1}{2}\|\delta G\|). \tag{6}$$

Then

$$R(\theta_{\mathrm{S}}^+) - R(\theta_{\mathcal{P}}^+) \geq \eta\mathcal{A}_{\mathcal{P}} - \eta^2\mathcal{B}_{\mathcal{P}}. \tag{7}$$

If $\mathcal{A}_{\mathcal{P}} > 0$, the risk is lower for $0 < \eta < \mathcal{A}_{\mathcal{P}}/\mathcal{B}_{\mathcal{P}}$ when $\mathcal{B}_{\mathcal{P}} > 0$, or for any admissible $\eta > 0$ when $\mathcal{B}_{\mathcal{P}} = 0$.

The beneficial sign depends on how the historical guide signals scale token gradients relative to task-risk gradients. Appendix B.4 separates overall gradient scale from token assignment and analyzes competitor attenuation; Appendix B.5 bounds the additional error from the cached top-K tail. The comparison takes two gradient steps from the same checkpoint and batch with a fixed historical reference. Fixed-prefix OPD teacher consistency is established separately in Appendix C.2.

4 Method

Section 3 shows when prior adaptation can align with a remaining correction. A present supervised loss does not reveal a token's prior change. REWIND measures that response by comparing current and frozen earlier predictions on the same prefix. The resulting historical guide signal calibrates subsequent supervision, while a demonstration or teacher supplies the correction (Figure 1).

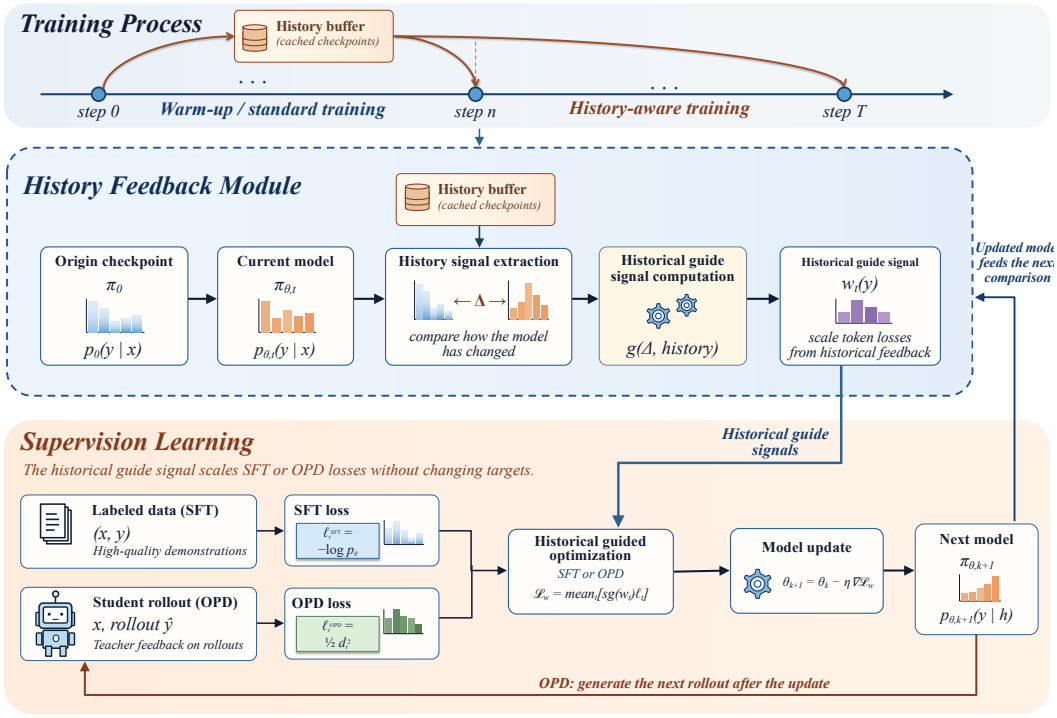


Figure 1: **REWIND training workflow.** Frozen and current predictions on matched prefixes provide a historical guide signal for existing SFT or OPD losses; only the student is updated.

4.1 EXISTING SUPERVISION

At prefix h_t and trained token y_t, write $p_\theta = \pi_\theta(y_t \mid h_t)$. SFT trains on demonstration tokens with $\ell_t^{\mathrm{SFT}} = -\log p_\theta$. OPD trains on student-generated prefixes and sampled tokens using a frozen teacher π_T; we use the sampled-token k2 surrogate of fixed-prefix reverse KL (Gu et al., 2024; Agarwal et al., 2024; Xu et al., 2025):

$$\ell_t^{\mathrm{OPD}} = \tfrac{1}{2} d_t^2, \qquad d_t = \log p_\theta - \log \pi_T(y_t \mid h_t). \tag{8}$$

The prefix and token are fixed during differentiation; the expected-gradient relation is derived in Appendix C.1. Both target-logit gradients contain the student's residual factor $1 - p_\theta$, multiplied by -1 in SFT and by the teacher discrepancy d_t in OPD. This shared factor provides an entry point for the learner's earlier response to calibrate subsequent supervision.

4.2 HISTORICAL RESPONSE AS A CALIBRATION SIGNAL

Reading the response to earlier updates. Let π_0, with parameters θ_0, be a frozen earlier checkpoint. It anchors the interval of measured change; the trainable student may start there or at a later checkpoint. The main comparisons use the run-start Base model as this reference. On the same prefix and token, let $p_0 = \pi_0(y_t \mid h_t)$. Their net change is

$$\Delta_t = \mathrm{logit}(p_\theta) - \mathrm{logit}(p_0), \qquad \mathrm{logit}(p) = \log \frac{p}{1-p}. \tag{9}$$

The sign of Δ_t records whether target odds rose or fell over the interval. It is the telescoped sum of intervening log-odds changes, yet requires only the endpoints. Equal current confidence can therefore conceal different responses to prior training.

Turning the response into a predictive reference. For a partition $\mathcal{P}_t$ of the vocabulary containing $\{y_t\}$, let $q_{r,t}(C) = \sum_{v \in C} \pi_r(v \mid h_t)$ for $r \in \{0, \theta\}$. We combine earlier and current predictions in

log-probability space to form a calibration reference:

$$\tilde{q}_t(C) = \frac{q_{\theta,t}(C)^{1-\lambda} q_{0,t}(C)^{\lambda}}{\sum_{C' \in \mathcal{P}_t} q_{\theta,t}(C')^{1-\lambda} q_{0,t}(C')^{\lambda}}, \qquad \lambda \in [0,1]. \tag{10}$$

Write $\tilde{p} = \tilde{q}_t(\{y_t\})$. On the binary partition, $\mathrm{logit}(\tilde{p}) = \mathrm{logit}(p_\theta) - \lambda\Delta_t$: $\lambda = 1/2$ places the reference halfway back in log-odds. Finer partitions also retain changes among competitors (Appendix A.4).

Historical guide signal. The reference supplies a target-coordinate residual, $1 - \tilde{p}$. We transfer its scale to the current loss through the positive signal

$$w_t^* = \frac{1-\tilde{p}}{1-p_\theta}, \qquad w_t = \mathrm{clip}(w_t^*, \kappa^{-1}, \kappa), \quad \kappa \geq 1. \tag{11}$$

Before clipping, $w_t^*(1-p_\theta) = 1-\tilde{p}$: the update remains at the current student while its student-side residual reflects prior change. On a binary partition, positive Δ_t gives $w_t > 1$ and negative Δ_t gives $w_t < 1$ when $\lambda > 0$ and $\kappa > 1$. Competitor changes also affect the signal on finer partitions (Theorem 3.1).

Partitions in the two settings. SFT caches the earlier checkpoint's target and top-K competitor scores on fixed demonstrations, pooling the rest into a tail. OPD keeps the sampled token separate, selects high-probability competitors from the pre-update student, and scores the frozen checkpoint on each rollout.

4.3 History-Calibrated Update

The historical guide signal acts on either original supervised loss:

$$L_w^{(s)} = \frac{\sum_t m_t \, \mathrm{sg}(w_t) \ell_t^{(s)}}{\sum_t m_t}, \qquad s \in \{\mathrm{SFT}, \mathrm{OPD}\}, \tag{12}$$

where m_t selects valid tokens. The positive, detached multiplier preserves each term's parameter-gradient direction while changing its contribution to the aggregate update.

For the unclipped weight, and policy-matched student probabilities in OPD, the shared target-logit factor becomes

$$\frac{\partial[\mathrm{sg}(w_t^*)\ell_t^{(s)}]}{\partial z_{y_t}} = \alpha_t^{(s)}(1-\tilde{p}), \qquad \alpha_t^{\mathrm{SFT}} = -1, \quad \alpha_t^{\mathrm{OPD}} = d_t. \tag{13}$$

The supervision coefficient remains intact: d_t still determines both the sign and part of an OPD correction's magnitude, while history adds a positive scale. Under clipping, the factor is $w_t(1-p_\theta)$ rather than exactly $1-\tilde{p}$. For SFT, Theorem 3.2 establishes a lower one-step task risk than the unweighted update under explicit alignment and step-size conditions. Appendix A.4 derives the target-coordinate relation.

4.4 Training Algorithm

In SFT, cached historical scores on fixed demonstrations are paired with current student predictions at each update. In OPD, the teacher and frozen checkpoint score each student rollout; pre-update student probabilities determine the historical guide signals before one student update. The next rollout uses the updated student. Algorithm 1 in Appendix A.5 gives the full procedure.

5 Experiments

We compare REWIND with SFT and OPD baselines at two model scales, Qwen3-4B-Base and Qwen3-1.7B-Base, on mathematics and code tasks. To keep the analysis comparable across settings, every REWIND arm uses the corresponding Base model as its frozen historical checkpoint. The exceptions are ablation A3, which replaces Base with intermediate training checkpoints, and the continuation diagnostics in Sections 5.5–5.6, which reuse its step-40 branch. The comparison arms, training settings, teachers, and evaluation protocol are specified in Appendix A.

5.1 MAIN RESULTS

Table 1: **Main results.** Mean@8 accuracy (%) on math and code benchmarks. Avg. is the unweighted mean of unrounded benchmark accuracies. All trained arms in this table use training seed 42 and the last matched checkpoint (Appendix A). The REWIND-SFT row uses Base as its frozen reference.

	Math (mean@8 ↑)							Code (mean@8 ↑)			
Model / method	MATH-500	AIME2025	AMC23	AIME24	Minerva	Olympiad	Avg.	HE+	MBPP+	LCB v5	Avg.
Direct checkpoint evaluation (frozen models)											
Qwen3-1.7B-Base	29.70	1.25	15.94	2.08	9.47	10.80	11.54	6.02	3.37	0.68	3.36
Qwen3-4B-Base	32.30	3.75	22.50	5.00	11.21	17.98	15.46	31.25	33.07	1.14	21.82
teacher	82.85	21.25	66.56	24.58	37.18	49.44	46.98	82.01	75.93	26.02	61.32
Supervised fine-tuning (SFT; Qwen3-4B-Base)											
SFT	61.45	14.17	43.44	11.67	27.21	29.04	31.16	69.05	66.80	23.27	53.04
InfoSFT	60.68	10.00	36.25	8.33	25.87	26.30	27.90$^{\downarrow 3.26}$	70.27	67.66	24.12	54.02$^{\uparrow 0.98}$
BLL-α1	65.15	8.75	**47.81**	10.00	26.79	**36.96**	32.58$^{\uparrow 1.42}$	74.01	68.32	25.11	55.81$^{\uparrow 2.77}$
EAFT	69.68	6.25	45.63	7.92	29.04	34.17	32.11$^{\uparrow 0.95}$	73.02	68.58	25.54	55.71$^{\uparrow 2.67}$
REWIND	**71.20**	**15.42**	46.88	**13.75**	**31.25**	36.20	**35.78**$^{\uparrow 4.62}$	**76.45**	**70.24**	**26.14**	**57.61**$^{\uparrow 4.57}$
Supervised fine-tuning (SFT; Qwen3-1.7B-Base)											
SFT	51.85	**2.92**	24.69	2.50	14.98	21.94	19.81	56.02	59.99	12.17	42.73
InfoSFT	58.28	2.50	25.63	3.75	21.60	24.11	22.64$^{\uparrow 2.83}$	57.32	60.78	12.51	43.54$^{\uparrow 0.81}$
BLL-α1	58.08	2.50	28.44	5.42	20.86	22.96	23.04$^{\uparrow 3.23}$	58.99	60.91	13.52	44.48$^{\uparrow 1.75}$
EAFT	57.63	2.08	**34.06**	7.08	18.61	24.17	23.94$^{\uparrow 4.13}$	58.54	60.35	13.27	44.05$^{\uparrow 1.32}$
REWIND	**59.03**	**2.92**	**34.06**	**7.92**	**21.83**	**24.65**	**25.07**$^{\uparrow 5.26}$	**59.83**	**62.07**	**14.35**	**45.42**$^{\uparrow 2.69}$
On-policy distillation (OPD; Qwen3-4B-Base)											
UNIFORM OPD	78.13	17.92	55.00	18.75	33.59	43.48	41.14	77.44	68.78	20.57	55.60
PW-OPSD	79.40	17.08	63.13	21.25	33.46	46.11	43.40$^{\uparrow 2.26}$	78.13	69.41	21.82	56.45$^{\uparrow 0.85}$
RENIO	79.80	17.08	56.56	20.42	34.56	45.61	42.34$^{\uparrow 1.20}$	77.97	69.21	22.16	56.45$^{\uparrow 0.85}$
FIRE-OPD	78.75	15.42	59.69	21.67	31.43	45.69	42.11$^{\uparrow 0.97}$	78.73	69.05	21.90	56.56$^{\uparrow 0.96}$
REWIND-OPD	**80.08**	**20.00**	**64.06**	**22.50**	**35.06**	**46.94**	**44.77**$^{\uparrow 3.63}$	**79.34**	**70.11**	**22.98**	**57.48**$^{\uparrow 1.88}$
On-policy distillation (OPD; Qwen3-1.7B-Base)											
UNIFORM OPD	59.03	4.17	33.44	4.58	23.90	24.06	24.86	53.43	59.09	9.20	40.58
PW-OPSD	61.03	5.00	**39.06**	4.58	23.16	26.41	26.54$^{\uparrow 1.68}$	55.87	60.05	9.83	41.92$^{\uparrow 1.34}$
RENIO	58.13	2.92	32.81	5.00	23.35	24.44	24.44$^{\downarrow 0.42}$	54.65	60.22	10.16	41.67$^{\uparrow 1.09}$
FIRE-OPD	62.85	5.00	36.56	5.42	23.16	27.33	26.72$^{\uparrow 1.86}$	54.34	59.52	8.96	40.94$^{\uparrow 0.36}$
REWIND-OPD	**63.43**	**5.83**	37.81	**6.25**	**23.99**	**27.76**	**27.51**$^{\uparrow 2.65}$	**57.16**	**61.01**	**10.97**	**43.05**$^{\uparrow 2.47}$

Supervised fine-tuning. REWIND achieves the highest average accuracy on both mathematics and code at both model scales in Table 1. Compared with standard SFT, it improves mathematical accuracy by 4.62 and 5.25 percentage points at 4B and 1.7B, respectively, with corresponding code gains of 4.57 and 2.69 points. At 4B, it leads four of the six mathematical benchmarks; at 1.7B, it matches or exceeds every evaluated baseline on all six. It also leads all three code benchmarks at both scales. These results support using historical predictive change to allocate demonstration-based supervision across model scales and task domains.

On-policy distillation. REWIND-OPD improves the use of teacher feedback on student-generated rollouts, achieving the highest mathematical and code averages among the compared OPD methods at both scales. Relative to uniform OPD, the mathematical average improves by 3.63 points at 4B and 2.65 points at 1.7B, while the code average improves by 1.88 and 2.47 points, respectively. It leads all six mathematical benchmarks at 4B, five of six at 1.7B, and all three code benchmarks at both scales. The mathematical OPD comparison uses one frozen DAPO-trained teacher and a matched training budget; the code results use the separate KodCode-trained teacher.

5.2 ABLATION OF THE HISTORICAL GUIDE SIGNAL

To identify how the historical guide signal drives REWIND's gains, we run four matched continuation ablations (Figure 2). We vary the token assignment and competitor information used to compute each signal, the fusion and allocation of historical information, the historical checkpoint, and the clipping range.

A1: Historical guide signal assignment improves accuracy. *Full* (A1-4) is REWIND-SFT: it computes each historical guide signal from the current and frozen Base predictions for the target. *Shuffle* permutes Full's clipped signals within width-0.1 bins of current target confidence, retaining their values but changing their token assignments. *Binary* compares the same Base and current states

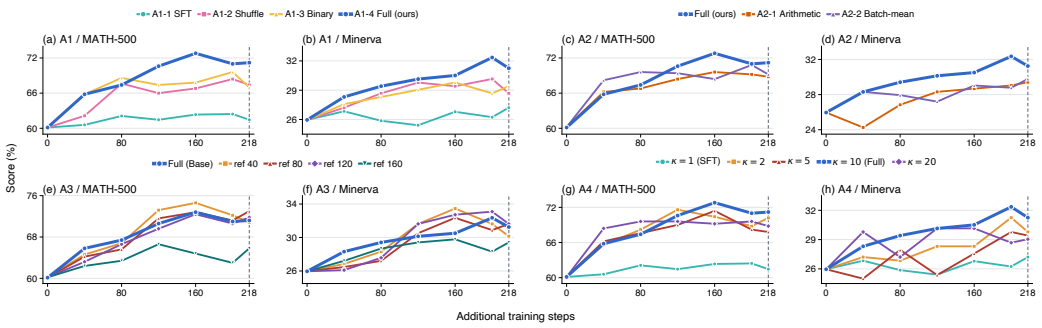


Figure 2: Results of ablations A1–A4 on MATH-500 and Minerva.

at each token but pools all non-target alternatives into one cell before computing its signal. At the final checkpoint in Figure 2(a)–(b), Full exceeds Shuffle by 3.80 and 2.57 points, and Binary by 4.20 and 1.84 points, on MATH-500 and Minerva respectively. The Full–Shuffle gap supports assigning historical guide signals to their corresponding tokens, beyond making the signals nonuniform; the Full–Binary gap supports retaining information about competing tokens.

A2: Ways to fuse historical information. We compare Full's normalized geometric reference with *Arithmetic-headroom*, which averages the Base and current target probabilities, and *Batch-mean*, which replaces Full's token-specific guide signals with one batch-level value. Both alternatives finish above matched uniform SFT on both benchmarks, indicating that change relative to Base remains useful under these tested fusion and allocation choices. In Figure 2(c)–(d), Full finishes highest on MATH-500 (71.20%, versus 68.80% and 69.20%) and Minerva (31.25%, versus 29.41% and 29.78%), with the competing scores from Arithmetic-headroom and Batch-mean in that order. Among these tested designs, the geometric fusion used by Full achieves the strongest results when its guide signals are applied to individual tokens.

A3: Choice of the historical checkpoint. All variants resume the same step-218 student and optimizer for 218 additional updates, varying the frozen reference between the run-start Base model and first-stage checkpoints at steps 40, 80, 120, and 160. At the fixed step-436 endpoint, every choice outperforms standard SFT on MATH-500 and Minerva (Figure 2(e)–(f)). The step-80 reference reaches 73.00% on MATH-500, compared with 71.20% using Base, and ties step 120 for the highest Minerva score (31.62%, versus 31.25% using Base). The later step-160 reference falls to 65.80% and 29.41%, respectively. Moving the reference later shortens the history interval, but the gains do not vary monotonically with that interval. Base remains the shared default in the main comparisons; selecting an intermediate reference for a new setting would require a separate validation rule.

A4: Sensitivity to the clipping bounds. We compare $\kappa \in \{1, 2, 5, 10, 20\}$, which sets the weight bounds to $[\kappa^{-1}, \kappa]$ in Eq. (11), while keeping all other settings fixed (Figure 2(g)–(h)). At $\kappa = 1$, every guide signal equals one and the method recovers uniform SFT; larger values permit stronger amplification and attenuation. Across the tested settings, all four nonuniform variants outperform uniform SFT on both benchmarks. The endpoint accuracy varies with κ, but the gain from historical guide signals persists over the tested nonuniform range.

5.3 Robustness Across Domains and Training Seeds

We test whether the SFT gains in Table 1 extend beyond mathematics and code and beyond a single training run (Table 2).

Gains extend to logical reasoning and instruction following. We train REWIND-SFT with domain-specific demonstrations (Appendix A) and evaluate it on logical reasoning (LogiQA2 (Liu et al., 2023a) and ReClor (Yu et al., 2020)) and instruction following (IFEval (Zhou et al., 2023) and IFBench (Pyatkin et al., 2025)) at both model scales (Table 2(a)). REWIND-SFT exceeds standard SFT on all four benchmarks at both scales, improving domain averages by 1.35–4.23 percentage points. It has the highest domain average among the compared methods in both domains at both scales.

Gains persist across training seeds. We repeat the matched Qwen3-4B-Base mathematical SFT comparison at training seeds 42, 1, and 2 (Table 2(b)). Seed 42 is the final checkpoint reported

for each method in Table 1. REWIND-SFT improves the six-benchmark average at every seed, and its three-run mean is higher on every benchmark. Panel (a) and the other baselines in Table 1 are reported from single training runs.

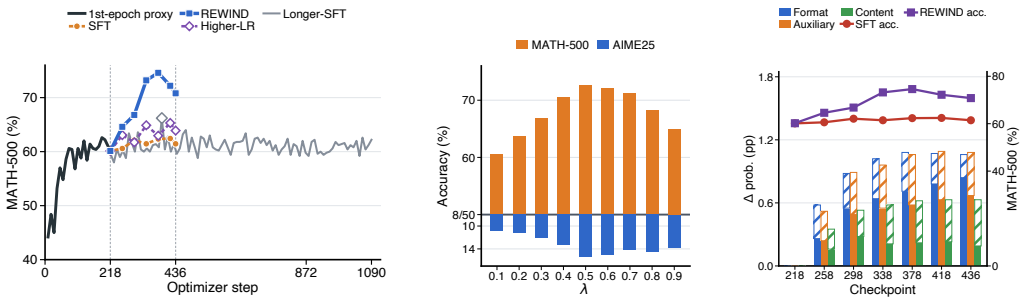
Table 2: **Robustness of REWIND-SFT across domains and training seeds.** All scores are mean@8 accuracy (%). Panel (b) reports the mean ± sample standard deviation over seeds 42, 1, and 2.

(a) Logical reasoning and instruction following

Method	Qwen3-1.7B-Base						Qwen3-4B-Base					
	Logic reasoning			Instruction following			Logic reasoning			Instruction following		
	LogiQA2	ReClor	Avg.	IFEval	IFBench	Avg.	LogiQA2	ReClor	Avg.	IFEval	IFBench	Avg.
SFT	43.81	47.30	45.56	62.92	23.50	43.21	66.13	74.55	70.34	79.81	26.25	53.03
InfoSFT	44.16	48.60	46.38	64.23	24.25	44.24	66.76	75.55	71.16	80.94	26.50	53.72
BLL	44.81	49.05	46.93	62.48	24.79	43.64	69.13	76.25	72.69	**81.54**	26.08	53.81
EAFT	44.35	48.25	46.30	63.75	**25.63**	44.69	68.31	77.58	72.95	80.43	27.25	53.84
REWIND-SFT	**45.18**	**49.63**	**47.41**	**65.50**	25.42	**45.46**	**70.36**	**78.78**	**74.57**	81.17	**27.58**	**54.38**

(b) Mathematical reasoning across training seeds (Qwen3-4B-Base).

Method	MATH-500	AIME2025	AMC23	AIME24	Minerva	Olympiad	Avg.
SFT	63.84 ± 2.14	12.92 ± 1.25	44.17 ± 2.43	12.92 ± 1.25	28.60 ± 1.45	30.25 ± 1.18	32.12 ± 1.23
REWIND-SFT	73.05 ± 1.64	16.39 ± 1.34	50.21 ± 3.15	15.83 ± 2.32	33.06 ± 1.62	37.30 ± 1.14	37.64 ± 1.61



(a) MATH-500 trajectories (b) Mixing coefficient (c) Continued fine-tuning

Figure 3: **Training trajectories and diagnostics on Qwen3-4B-Base.** (a) MATH-500 training trajectories from the shared step-218 checkpoint. (b) mean@8 accuracy across λ (bar baselines: 50% for MATH-500, 8% for AIME25). (c) Continued-fine-tuning target-probability gains by token class (solid: SFT; hatched: additional REWIND gain) and MATH-500 accuracy.

5.4 SENSITIVITY TO HISTORICAL MIXING

We test how much of the run's history should shape the reference by varying the mixing coefficient λ in Eq. (10) for REWIND-SFT on Qwen3-4B-Base. The main comparisons in Table 1 fix $\lambda = 0.5$; Figure 3(b) examines its sensitivity in a post-hoc sweep. At $\lambda = 0$, the reference equals the current prediction and every historical guide signal equals one. Increasing λ therefore changes how strongly the frozen checkpoint influences the allocation of the original SFT loss. As λ decreases from 0.9 to 0.5, accuracy improves overall, from 13.75% to 15.42% on AIME25 and from 65.03% to 72.63% on MATH-500. Below 0.5, accuracy falls more sharply, reaching 10.83% and 60.53% at $\lambda = 0.1$. Values in $[0.4, 0.7]$ remain comparatively strong on both benchmarks, while performance decreases toward the tested extremes. The symmetric default has the highest accuracy among the nine displayed coefficients on both benchmarks, with a comparatively strong range around it.

5.5 Beyond Longer SFT and a Higher Learning Rate

We compare REWIND-SFT with two optimization controls from the shared step-218 checkpoint. Higher-LR increases the standard SFT learning rate fivefold, from 10^{-5} to 5×10^{-5}, with all other settings fixed. Longer-SFT extends standard SFT to five epochs, with all other training and evaluation settings unchanged. At step 436, the REWIND-SFT branch with checkpoint 40 as its frozen reference reaches 70.8% on MATH-500, versus 61.5% for standard SFT (Figure 3(a)). Table 1's 71.20% is the separate Base-reference REWIND-SFT arm at the same step. Even at their best recorded checkpoints, Higher-LR reaches 65.3% and Longer-SFT reaches 66.2%: the REWIND endpoint exceeds these controls by 5.5 and 4.6 points, respectively. Together with the shuffled-weight control in A1 (Figure 2(a)–(b)), these comparisons support assigning historical guide signals to their corresponding tokens.

5.6 Prediction Changes During Continued Fine-Tuning

We compare teacher-forced target probabilities on 10,000 fixed AM-Math completions (24,846,055 supervised positions). Standard SFT and REWIND-SFT both continue from checkpoint 218 for 218 updates; REWIND uses checkpoint 40 as its frozen reference, while plotted changes are measured from checkpoint 218. Fixed lexical rules group positions as format, auxiliary/function, or normal content (Appendix A.3).

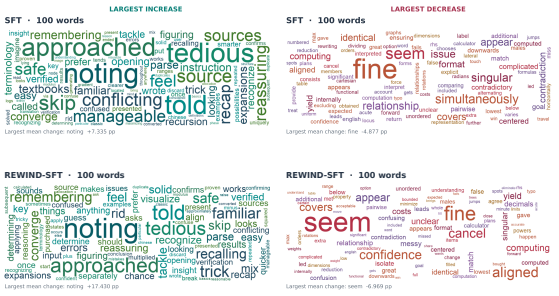


Figure 4: Lexical target-probability changes from checkpoint 218 to 338.

Across the six checkpoints, REWIND's mean target-probability gain on normal-content tokens is approximately 2.6× SFT's (Figure 3(c)), while its MATH-500 accuracy is higher by 8.64 points on average over the same grid. The lexical response is selective: 84.5% of 600 retained verb forms gain under REWIND, versus 80.0% under SFT, whereas *seem* and *unclear* decline more under REWIND (Figure 4). Together, these trends distinguish history-guided weighting from a uniform increase in demonstration fit: larger gains on normal content and opposing lexical shifts accompany higher held-out accuracy.

5.7 Training Cost

REWIND's additional computation is a forward-only evaluation of the frozen historical checkpoint: SFT caches its scores once on fixed demonstrations, whereas OPD scores each new rollout online. For Qwen3-4B-Base mathematical SFT on eight H20 GPUs, building the $K = 100$ cache takes 1.00 hour and stores 202.43 GiB of reference data (Appendix A.2). The matched 218-step continuation takes 5.71 versus 5.66 hours for standard SFT (0.9% extra; 94.29 versus 93.69 seconds per step). Including the shared first epoch and cache gives 12.25 versus 11.20–11.21 hours (9.3–9.4% extra). In this same continuation, MATH-500 accuracy rises from 61.5% to 70.8% at step 436, a 9.3-point gain (Section 5.5). For OPD, the matched 144-step tail after the shared step-120 prefix takes 7.21 versus 6.33 hours (13.8% extra); frozen-reference scoring accounts for 46.6 of the 52.6 added minutes, while actor-update throughput remains nearly unchanged. Including the shared prefix gives 12.36 versus 11.48 hours (7.6% extra logged step time, summed from completed segments rather than measured in one launch-to-finish run). In the corresponding 4B mathematical OPD comparison, the six-benchmark average rises from 41.14 to 44.77, a 3.63-point gain (Table 1). Thus, in these measured settings, the additional frozen-reference pass offers a favorable accuracy–time trade-off: substantial improvements for a modest increase in wall-clock time.

6 Conclusion

We introduced REWIND, a simple yet effective way to use training history to allocate existing supervision. Its benefits span SFT and OPD, two Qwen3 model scales, and multiple task domains. Further discussion appears in Appendix D.

AI USE STATEMENT

In this work, generative AI tools were used to assist with refining research ideas and experimental designs, checking and refining mathematical proofs, literature search, code development, figure preparation, and language editing and translation of the manuscript. All mathematical arguments were independently verified by the authors. AI-assisted code was reviewed and tested by the authors for correctness, and all AI-assisted textual edits were reviewed before inclusion in the manuscript. The authors take full responsibility for the final content of this work, including text, claims, code, figures, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide anonymous supplementary code containing a reference implementation of the core components of REWIND for SFT and OPD. Appendix A reports the models, datasets, training and evaluation configuration, and implementation settings; Appendix A.4 and Algorithm 1 describe the reference construction and the training procedure; and Appendices B and C provide the assumptions and proofs for the theoretical results, including the OPD gradient and teacher-consistency derivations.

REFERENCES

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

Alnur Ali, J. Zico Kolter, and Ryan J. Tibshirani. A continuous-time view of early stopping for least squares regression. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1370–1378. PMLR, 2019. URL https://proceedings.mlr.press/v89/ali19a.html.

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

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2306.08543; later arXiv title: "MiniLLM: On-Policy Distillation of Large Language Models".

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. arXiv:2402.14008.

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

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. *arXiv preprint arXiv:2403.07974*, 2024.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2402.03898.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay 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. *arXiv preprint arXiv:2206.14858*, 2022.

Gaotang Li, Ruizhong Qiu, Xiusi Chen, Heng Ji, and Hanghang Tong. Beyond log likelihood: Probability-based objectives for supervised fine-tuning across the model capability continuum. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026a. arXiv:2510.00526.

Yuying Li, Leqi Zheng, Yongzi Yu, Wenrui Zhou, Xuchang Zhong, Xing Hu, Jing Jin, Hangjie Yuan, and Tao Feng. Filter, then reweight: Rethinking optimization granularity in on-policy distillation. *arXiv preprint arXiv:2606.02684*, 2026b.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Chen Lin, Kedi Chen, and Wei Zhang. ReNIO: Reweighting negative trajectory importance for LLM on-policy distillation. *arXiv preprint arXiv:2606.23104*, 2026.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Not all tokens are what you need for pretraining. In *Advances in Neural Information Processing Systems (NeurIPS)*,

2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/3322a9a72a1707de14badd5e552ff466-Abstract-Conference.html. arXiv:2404.07965.

Hanmeng Liu, Jian Liu, Leyang Cui, Zhiyang Teng, Nan Duan, Ming Zhou, and Yue Zhang. LogiQA 2.0—an improved dataset for logical reasoning in natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2947–2962, 2023a. doi: 10.1109/TASLP.2023.3293046.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. *arXiv preprint arXiv:2305.01210*, 2023b.

Xiaogeng Liu, Xinyan Wang, Yingzi Ma, Yechao Zhang, and Chaowei Xiao. When are teacher tokens reliable? position-weighted on-policy self-distillation for reasoning. *arXiv preprint arXiv:2605.21606*, 2026.

Sören Mindermann, Jan M. Brauner, Muhammed T. Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez, Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning (ICML)*, volume 162, pp. 15630–15649. PMLR, 2022. URL https://proceedings.mlr.press/v162/mindermann22a.html. arXiv:2206.07137.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022. URL https://proceedings.neurips.cc/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

Keqin Peng, Chen Li, Yuanxin Ouyang, Yancheng Yuan, and Liang Ding. Diagnosing and mitigating thinking collapse in on-policy self-distillation. *arXiv preprint arXiv:2607.10805*, 2026.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. Generalizing verifiable instruction following. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2507.02833.

Mahdi Sabbaghi, George Pappas, Adel Javanmard, and Hamed Hassani. Infosft: Learn more and forget less with information-aware token weighting. *arXiv preprint arXiv:2605.14967*, 2026.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems (EuroSys)*, 2025.

Yongliang Wu, Yizhou Zhou, Ziheng Zhou, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. On the generalization of SFT: A reinforcement learning perspective with reward rectification. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/2e53c02ea028cbf603f4b6b47fef3d97-Abstract-Conference.html.

Hongling Xu, Qi Zhu, Heyuan Deng, Jinpeng Li, Lu Hou, Yasheng Wang, Lifeng Shang, Ruifeng Xu, and Fei Mi. KDRL: Post-training reasoning LLMs via unified knowledge distillation and reinforcement learning. *arXiv preprint arXiv:2506.02208*, 2025. URL https://arxiv.org/abs/2506.02208.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Fangxu Yu, Weijia Xu, Michael Xu, Tianyi Zhou, and Zinan Lin. Weak-to-strong on-policy distillation. *arXiv preprint arXiv:2607.26246*, 2026. URL https://arxiv.org/abs/2607.26246.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. ReClor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/2002.04326.

Han Zhao, Haotian Wang, Yiping Peng, Sitong Zhao, Xiaoyu Tian, Shuaiting Chen, Yunjie Ji, and Xiangang Li. 1.4 million open-source distilled reasoning dataset to empower large language model training. *arXiv preprint arXiv:2503.19633*, 2025. URL https://arxiv.org/abs/2503.19633.

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

A EXPERIMENTAL DETAILS

Comparison arms. The SFT comparison includes standard SFT, InfoSFT (Sabbaghi et al., 2026), BLL-$\alpha 1$ (Li et al., 2026a), EAFT (Diao et al., 2026), and REWIND-SFT. The OPD comparison includes uniform OPD, PW-OPSD (Liu et al., 2026), ReNIO (Lin et al., 2026), FiRe-OPD (Li et al., 2026b), and REWIND-OPD.

Rewind parameters. The main comparisons use rollback strength $\lambda = 1/2$ and $\kappa = 10$, clipping historical guide signals to $[0.1, 10]$ in SFT and OPD. Section 5.4 reports a λ sweep for Qwen3-4B-Base SFT runs per coefficient.

Training settings. Every SFT arm uses a learning rate of 10^{-5}, and every OPD arm uses 10^{-6}. REWIND-SFT matches standard SFT hyperparameters; the other SFT baselines use their method-specific settings aside from this shared learning rate. The OPD baselines follow the method-specific settings reported in their papers, aside from this shared learning rate, and are implemented in the `verl` framework. Within each matched comparison, arms share the data or prompt pool, seed, epoch and optimizer-step budgets. For Table 1, each trained arm is reported at the last checkpoint of the matched training budget. Every method saves a checkpoint every 40 steps. The run registry records which evaluation decisions were fixed in advance and which analyses were specified after results were available. The SFT training-sequence limits are $4{,}096$ tokens for mathematics, code, and logical reasoning, and 512 for instruction following. For mathematics OPD, prompt and sampled-response limits are $1{,}024$ and $7{,}168$ tokens, respectively.

Models. The main comparison uses Qwen3-4B-Base and Qwen3-1.7B-Base (Yang et al., 2025) as student backbones. The main SFT comparisons initialize training from the corresponding Base checkpoint, which also serves as REWIND's frozen reference. All OPD arms also start from the corresponding Base model. The other OPD methods apply their respective objectives from the start; REWIND-OPD uses standard OPD for the first 0.5 epoch and then activates its historical guide signals for the remaining training budget. Its frozen historical reference remains Base throughout. In the SFT continuation analyses, the student instead resumes from step 218 while the reference remains frozen at Base or, in A3 and the diagnostics that reuse its step-40 branch, at an earlier checkpoint. The mathematical OPD arms, at both student scales, use a frozen Qwen3-4B-Base-GRPO teacher trained from Qwen3-4B-Base on DAPO-Math using GRPO in `verl` v0.8 (Sheng et al., 2025). All code OPD arms use a separate frozen Qwen3-4B-Base GRPO teacher trained on KodCode-Light-RL-10K. The first block of Table 1 reports the two Base checkpoints and the domain-specific teacher references.

Datasets. Mathematical SFT uses the mathematics subset of AM-DeepSeek-R1-1.4M (Zhao et al., 2025): $111{,}657$ examples with full reasoning traces, consumed for two epochs by each mathematical SFT arm. Code SFT uses the execution-verified code subset of the same AM corpus. Logical-reasoning SFT merges the LogiQA2 and ReClor training data, uses Qwen3-32B in thinking mode to generate responses, and retains answer-verified responses as demonstrations. The standard instruction-following SFT controls use $11{,}335$ verifier-passing demonstrations from the released IF multi-constraint SFT corpus. For OPD, all mathematical arms use prompts from DAPO-Math-17K-Processed, and all code arms use prompts from KodCode-Light-RL-10K.

Benchmarks. Mathematical reasoning is evaluated on six benchmarks: MATH-500 (Hendrycks et al., 2021; Lightman et al., 2023), the 500-problem subset of MATH; AIME24 and AIME2025, the 30 problems of the 2024 and 2025 American Invitational Mathematics Examinations; AMC23, 40 problems from the 2023 American Mathematics Competitions; Minerva Math (Lewkowycz et al., 2022), 272 undergraduate-level problems; and OlympiadBench (He et al., 2024), the 675-problem open-ended, text-only English mathematics subset. Code is evaluated on HumanEval+ and MBPP+ (Liu et al., 2023b), with 164 and 378 problems, and LiveCodeBench v5 (Jain et al., 2024), with 880 problems.

Evaluation settings. All trained arms are evaluated under one decoding configuration. For each question we draw $n{=}8$ samples in a single sampling call at temperature 0.5, top-$p = 1.0$, and seed 42, and report mean@8, the accuracy averaged over the eight samples. Generation is capped at

8,192 new tokens for AIME24, AIME2025, and AMC23; 4,096 for MATH-500, Minerva Math, OlympiadBench, all code benchmarks, LogiQA2, and ReClor; and 1,280 for IFEval and IFBench.

A.1 Implementation settings

SFT cache. We cache the target probability, the $K = 100$ highest-probability non-target tokens and their probabilities, and the remaining tail mass on fixed demonstration prefixes. The current student is coarsened onto this cached partition. Its tail mass is $1 - p_\theta - \sum_{v \in \text{top-}K} \pi_\theta(v \mid h_t)$. Probabilities are computed in FP32; the student tail is clamped at zero, and the reference tail is floored at the smallest positive FP32 normal value before caching its logarithm. The geometric interpolation in Eq. (10) defines the reference when a tail cell rounds to zero. On one completed Qwen3-4B-Base mathematical SFT run, the frozen reference's cache covers 111,657 samples and 269,003,962 supervised tokens. Each candidate stores a 4-byte token id and a 4-byte log-probability, and the target and tail each add 4 bytes, so a $K = 100$ record is 808 bytes. At this token count the cache is 202.43 GiB. Building this $K = 100$ cache offline on 8 H20 GPUs took 60 minutes and 8 seconds. Training reads the cache, and the reference model receives no forward pass.

OPD protocol. The sampler and all scoring and update forwards use temperature $T = 1$, with one optimizer step per rollout batch. At each rollout position, the pre-update student selects its $K = 15$ highest-probability tokens. The sampled token and the other selected tokens form singleton cells; the remaining vocabulary mass is one tail. We score the frozen historical checkpoint on this partition. Stored pre-update student cell masses determine the historical guide signals, while the base loss uses differentiable current-student probabilities on the sampled token. These probabilities are policy-matched before the update, satisfying the condition in Eq. (13); no importance ratio is applied. The expected reverse-KL gradient and the potential in Appendix C.2 use samples from the full student distribution.

A.2 Training cost on Qwen3-4B-Base mathematics

We report measured time and storage separately from model-equivalent compute. SFT scores its frozen reference once on fixed demonstrations; OPD scores it during each rollout batch. Table 3 groups both settings while marking their different timing boundaries.

SFT: offline reference and update throughput. The SFT rows use eight H20 GPUs and the 4,096-token completion-only mathematical corpus with global batch 512. The two standard-SFT continuations share the step-218 model and optimizer state. The frozen Base cache at $K = 100$ is built once on eight H20 GPUs in 60 minutes and 8 seconds and occupies 202.43 GiB.

Table 3: **4B mathematical training cost.** SFT uses eight H20 GPUs; OPD uses four actor and four teacher H20 GPUs. SFT time is Trainer runtime, and OPD time sums logged steps including checkpoint saves. Rates are model-arithmetic $6P$ estimates in TFLOP/s/GPU. Mean step time divides unrounded row time by the number of updates.

Stage	Steps	Time (h)	Mean (s/step)	SFT rate	OPD actor rate
SFT (8 H20 GPUs)					
Shared standard SFT	1–218	5.54	91.47	40.70	—
Standard SFT continuation	219–436	5.66	93.69	39.68	—
REWIND-SFT ($K = 100$)	219–436	5.71	94.29	39.31	—
OPD (4 actor + 4 teacher H20 GPUs)					
Shared uniform-OPD prefix	1–120	5.15	154.56	—	—
Uniform OPD tail	121–264	6.33	158.31	—	13.17
REWIND-OPD tail	121–264	7.21	180.22	—	13.20

The rate columns use different denominators. SFT divides $6P$ times supervised tokens by eight GPUs and Trainer runtime; OPD divides $6P$ times logged sequence tokens by four actor GPUs and actor-update time. The OPD rate describes the actor update, not generation or teacher and historical-reference scoring, so it is not a whole-pipeline speed ratio against SFT.

The two standard-SFT epochs take 11.20–11.21 hours in total. The measured $K = 100$ continuation takes 5.71 hours for 218 updates. Together with the shared first epoch, this gives 11.25 hours of training, or 12.25 hours including the one-time frozen-reference cache. A separate 25-step $K = 100$ probe used a fresh optimizer and no intermediate checkpoint saves, so its timing is not substituted for the completed continuation.

OPD: matched logged-step timing. Uniform OPD and REWIND-OPD share the step-120 checkpoint. The OPD rows in Table 3 sum the 144 recorded step timers over updates 121–264; these include checkpoint saves but exclude process startup and shutdown.

The matched tails differ by 0.877 hours (52.6 minutes, 13.8%); historical-reference scoring accounts for 46.6 minutes, or 88.6% of that difference. The recorded uniform-OPD prefix (steps 1–120) takes 5.15 hours. Adding this same prefix to each tail gives 11.48 hours for uniform OPD and 12.36 hours for the shared-prefix REWIND-OPD schedule, a 7.6% increase in logged update time. This full-schedule figure combines completed segment logs rather than a separate launch-to-finish REWIND-OPD run. The tails differ by 1.3% in logged tokens, while their actor-update rates are nearly equal.

Model-equivalent compute. For $P = 4.023$ billion parameters, the 269.0 million first-epoch and 267.8 million second-epoch supervised tokens correspond to 12.96 EFLOPs of SFT at $6P$ per token. The one-time frozen-reference pass adds 2.16 EFLOPs at $2P$ per token (16.7%). For OPD, actor rollout, old-policy scoring, teacher scoring, and actor update total $12P$ per sequence token; REWIND-OPD adds a $2P$ historical-reference score on its continuation. The logged tokens correspond to 2.06 EFLOPs for uniform OPD and 2.29 EFLOPs for REWIND-OPD including the shared prefix (11.1% more), of which 0.212 EFLOPs come from reference scoring. These estimates exclude attention's sequence-length term, unsupervised-prefix work, padding, cache effects, and non-model kernels.

A.3 TOKEN-LEVEL ANALYSIS PROTOCOL

The comparison uses standard SFT and the A3-1 REWIND-SFT continuation. Both continue from checkpoint 218 for 218 additional updates. Checkpoint 218 is the baseline for measuring probability changes. The cross-checkpoint summaries in Section 5.6 use the six post-baseline checkpoints 258, 298, 338, 378, 418, and 436. For each method, we average the normal-content probability gains relative to checkpoint 218, then take the ratio of those averages. This uses the total REWIND gain, equal to the solid bar plus its hatched extension. The reported MATH-500 mean gap averages the paired accuracy differences at the same six checkpoints.

For Figure 3(c), all models score the same teacher-forced prefixes, so each target probability refers to an identical token and context across checkpoints. Fixed lexical rules assign every valid supervised position to one of three mutually exclusive categories. Formatting tokens comprise protocol markers, whitespace, and non-mathematical punctuation. Auxiliary/function tokens match a fixed word list containing articles, pronouns, auxiliary verbs, prepositions, conjunctions, and other function words. Normal content comprises all remaining tokens, including ordinary text, numbers, and mathematical expressions. The categories contain 4,077,436 (16.41%), 4,414,540 (17.77%), and 16,354,079 (65.82%) supervised positions, respectively; percentages use all 24,846,055 supervised positions as the denominator. Mean target probabilities for format, auxiliary/function, and normal-content tokens are (0.7988, 0.6518, 0.8212) at Base and (0.8287, 0.6877, 0.8444) at checkpoint 218. We average target-token probabilities over positions within each category and report changes relative to checkpoint 218, the shared continuation point. For the lexical comparison at checkpoint 338, a contextual part-of-speech filter retains verb occurrences outside auxiliary and formula spans. We keep word forms appearing at least 100 times in at least 20 completions, yielding 600 verb forms; the reported proportions count forms with a positive mean target-probability change from checkpoint 218. On the same fixed prefixes, *seem* and *unclear* decrease by 6.97 and 2.15 percentage points under REWIND-SFT, compared with 2.86 and 1.03 under SFT (Figure 4).

A.4 REFERENCE CONSTRUCTION AND COMPETITOR CORRECTION

For a fixed prefix and token, let $p_t^{[j]}$ be the target probability at the jth state of a chosen reference-to-current sequence, with $p_t^{[0]} = p_0$ and $p_t^{[J]} = p_\theta$. The local index $[0]$ labels the selected historical

reference, irrespective of its global training step. Successive changes telescope:

$$\sum_{j=1}^{J}\Big[\operatorname{logit}(p_t^{[j]}) - \operatorname{logit}(p_t^{[j-1]})\Big] = \operatorname{logit}(p_\theta) - \operatorname{logit}(p_0) = \Delta_t.$$

Geometric construction. On a fixed partition containing the target singleton, define

$$q_{r,t}(C) = \sum_{v\in C}\pi_r(v \mid h_t), \qquad C \in \mathcal{P}_t, \quad r \in \{0,\theta\}. \tag{14}$$

Writing $\delta_{t,C} = \log q_{\theta,t}(C) - \log q_{0,t}(C)$, reversing a fraction λ of this change gives $q_{\theta,t}(C)e^{-\lambda\delta_{t,C}} = q_{\theta,t}(C)^{1-\lambda}q_{0,t}(C)^{\lambda}$. Normalizing yields Eq. (10). On the binary partition,

$$\operatorname{logit}(\tilde{p}) = \operatorname{logit}(p_\theta) - \lambda\Delta_t = (1-\lambda)\operatorname{logit}(p_\theta) + \lambda\operatorname{logit}(p_0). \tag{15}$$

This is a midpoint of predictions when $\lambda = 1/2$, without assuming that neural-network parameters follow a midpoint path.

Target-logit calibration. For $\lambda < 1$, the auxiliary loss $\ell_t^{\mathrm{cal}} = -\log\tilde{p}/(1-\lambda)$ has target-logit derivative $-(1-\tilde{p})$ under a fixed partition. Matching this residual to the weighted SFT residual $-w_t^*(1-p_\theta)$ gives Eq. (11); its ratio also defines the $\lambda = 1$ endpoint directly. The base losses share the student-side factor

$$\frac{\partial \ell_t^{(s)}}{\partial z_{y_t}} = \alpha_t^{(s)}(1-p_\theta), \qquad \alpha_t^{\mathrm{SFT}} = -1, \quad \alpha_t^{\mathrm{OPD}} = d_t. \tag{16}$$

The equality in Eq. (13) concerns the trained-token coordinate and requires the weighting and differentiable student probabilities to match in OPD. With clipping, the calibrated factor is $w_t(1-p_\theta)$; the full parameter gradient of each term remains a positive scalar multiple of its unweighted gradient. Near saturation, the raw ratio may be large although the SFT target-logit gradient $1-\tilde{p}$ remains below one, so clipping limits relative amplification.

Theorem 3.1 characterizes the normalized geometric reference for arbitrary fixed partitions and clipped weights. Its proof in Appendix B.2 derives the conditional-overlap identity (18) and the monotone refinement bound (5). Appendix B.5 additionally bounds the weight, update, and one-step risk error of retaining a top-K head and merging the remaining vocabulary into one tail cell. This bound applies directly to the cached $K = 100$ SFT partition and to the online student top-15 OPD partition.

A.5 Training workflow

SFT. Algorithm 1 first caches the frozen checkpoint's predictions on fixed demonstration prefixes. Each batch combines current and cached scores into detached historical guide signals for the original SFT losses; optimization updates the student using the cache.

OPD. In `verl` (Sheng et al., 2025), the student is the actor and the frozen checkpoint supplies scores through the reference-policy input. For each student rollout, the teacher and reference score the same trajectories; pre-update actor probabilities and reference scores define detached historical guide signals for the teacher-distillation losses. The actor takes one weighted update before generating the next rollout. Numerical settings are given in Appendix A.1.

Algorithm 1 REWIND with cached SFT or online OPD scoring

Require: Student initialization π_{init}; frozen historical reference π_0 or its cache; data/prompt source $\mathcal{D}$; setting s; frozen teacher π_T for OPD; partition rule; rollback strength $\lambda \in [0, 1]$; clip $\kappa \geq 1$

Initialize $\pi_\theta \leftarrow \pi_{\text{init}}$
for each update **do**
  if $s = \text{SFT}$ **then**
    Read demonstration batch (h_t, y_t) and cached $(\mathcal{P}_t, q_{0,t})$
    Set $q_{\text{run},t} \leftarrow q_{\theta,t}$ on the cached cells
  else
    Sample rollout; set $y_t \leftarrow a_t$ and select the student top-15 tokens
    Form $\mathcal{P}_t$ from $\{a_t\}$, other top-15 singletons, and a tail
    Store pre-update $q_{\text{b},t}$ and score $q_{0,t}$ on $\mathcal{P}_t$
    Set $q_{\text{run},t} \leftarrow q_{\text{b},t}$
  end if
  Compute differentiable base losses $\ell_t^{(s)}$ and valid-token masks m_t
  Form $\tilde{p}$ from $q_{\text{run},t}, q_{0,t}$ using Eq. (10)
  Set $p_{\text{run},t} \leftarrow q_{\text{run},t}(\{y_t\})$
  Set $w_t \leftarrow \text{sg}\Big(\text{clip}\Big(\frac{1-\tilde{p}}{1-p_{\text{run},t}}, \kappa^{-1}, \kappa\Big)\Big)$
  Update only θ with Eq. (12)
end for

A.6 SYSTEM PROMPT FOR MATHEMATICS AND CODE

Shared system message

```
You are a helpful assistant.  To answer the user's question, you
first think about the reasoning process and then provide the user
with the answer.  The reasoning process and answer are enclosed
within <think> and <answer> tags, respectively, i.e., <think>
reasoning process here </think> <answer> answer here </answer>.
```

B PROOFS FOR HISTORICAL INFORMATION AND THE IMPLEMENTED UPDATE

Scope and notation. The results below analyze detached historical guide signals on fixed finite partitions with a target singleton, fixed $\lambda \in [0, 1]$, and clipping at $[\kappa^{-1}, \kappa]$. Logits may depend nonlinearly on shared parameters, and probabilities are treated as exact and strictly positive. The expansion around the selected reference is local; the weighting identities and finite-step risk comparisons apply at arbitrary checkpoints. The quadratic statistical example starts its trajectory at the reference.

B.1 HISTORICAL DISPLACEMENT AND THE REMAINING CORRECTION

Proof of Proposition 3.1. Put $H = \text{diag}(\gamma_j)$. The empirical field is $G_{\text{S}}^{\text{quad}}(u) = H(u - \mu - \xi)$. Solving $\dot{u} = -G_{\text{S}}^{\text{quad}}(u)$ from zero gives $u_j(\tau) = \rho_j(\tau)(\mu_j + \xi_j)$, $\rho_j(\tau) = 1 - e^{-\gamma_j \tau}$. For $\rho = \rho_j(\tau)$,

$$u_j(\mu_j - u_j) = \rho(\mu_j + \xi_j)[(1 - \rho)\mu_j - \rho\xi_j].$$

Taking expectations gives $A_j = \rho(1 - \rho)\mu_j^2 - \rho^2 v_j$. For $\tau > 0$, $0 < \rho < 1$, so positive alignment is equivalent to $(1 - \rho)\mu_j^2 > \rho v_j$. This gives the stated threshold when $\mu_j \neq 0$. A pure-noise direction has $A_j = -\rho^2 v_j \leq 0$. The result allows correlated coordinates of ξ. □

B.2 EXACT MULTICLASS WEIGHTS, REFINEMENT, AND CLIPPING

For one token suppress i, and write $p = P_\theta(y)$, $q = P_0(y)$. For a partition $\mathcal{P}$, put

$$A = q^\lambda p^{1-\lambda}, \qquad B_{\mathcal{P}} = \sum_{C \neq \{y\}} P_0(C)^\lambda P_\theta(C)^{1-\lambda}.$$

The reference target probability is $f_{\mathcal{P}} = A/(A + B_{\mathcal{P}})$.

Proof of Theorem 3.1. Factoring out the non-target masses gives $B_{\mathcal{P}} = (1-q)^{\lambda}(1-p)^{1-\lambda}c_{\mathcal{P}}$. Consequently,

$$\frac{f_{\mathcal{P}}}{1-f_{\mathcal{P}}} = \frac{1}{c_{\mathcal{P}}}\left(\frac{q}{1-q}\right)^{\lambda}\left(\frac{p}{1-p}\right)^{1-\lambda}. \tag{17}$$

Hölder's inequality gives $0 < c_{\mathcal{P}} \leq 1$, and hence

$$\operatorname{logit}(f_{\mathcal{P}}) = \operatorname{logit}(p) - \lambda\Delta + D_{\mathcal{P}}, \qquad D_{\mathcal{P}} = -\log c_{\mathcal{P}}. \tag{18}$$

Setting $s = \lambda\Delta - D_{\mathcal{P}}$ yields

$$\frac{1-f_{\mathcal{P}}}{1-p} = \frac{1}{1-p+pe^{-s}}.$$

Clipping gives Eq. (4). For $\kappa > 1$, clipping retains which side of one the raw weight lies on. The raw ratio exceeds one exactly when $s > 0$.

If a non-target cell C is split into cells C_b, Hölder gives

$$\sum_b P_0(C_b)^{\lambda}P_\theta(C_b)^{1-\lambda} \leq \left(\sum_b P_0(C_b)\right)^{\lambda}\left(\sum_b P_\theta(C_b)\right)^{1-\lambda}.$$

Thus $c_{\mathcal{Q}} \leq c_{\mathcal{P}}$ and $d = D_{\mathcal{Q}} - D_{\mathcal{P}} \geq 0$ for a refinement $\mathcal{Q}$. Let $w^*(s) = (1-p+pe^{-s})^{-1}$. Directly,

$$e^{-d} \leq \frac{w^*(s-d)}{w^*(s)} = \frac{1-p+pe^{-s}}{1-p+pe^{-s+d}} \leq 1.$$

Clipping in positive coordinates is clipping of the logarithm followed by exponentiation. Scalar clipping is monotone and nonexpansive, so

$$0 \leq \log W_p(s) - \log W_p(s-d) \leq d.$$

Exponentiation proves Eq. (5), including cases where one or both weights are clipped. At $\lambda = 0$ or 1, every conditional overlap is one, and the reference equals the current or historical-checkpoint prediction, respectively. At $\kappa = 1$ all weights are one. Finally, stop-gradient gives $\nabla_\theta[\operatorname{sg}(w_i)\ell_i] = w_i g_i$ regardless of the logit parameterization. □

Equality and interpretation. For $0 < \lambda < 1$, a split leaves its geometric mass unchanged exactly when the two endpoint distributions agree conditionally within that cell. Otherwise refinement strictly lowers the raw weight, although clipping can make the clipped weights equal. In a refined partition, competitor disagreement can offset a positive target-probability change because the weight depends on $\lambda\Delta - D$.

B.3 Nonlinear parameter-space feedback

For the full multiclass margin $m_i(\theta) = \operatorname{logit}(p_i(\theta))$, set $b_i = \nabla m_i$ and $g_i = -(1-p_i)b_i$.

Lemma B.1 (Exact feedback at an arbitrary checkpoint)**.** *Put $s_i = \lambda\Delta_i - D_i^{\mathcal{P}}$ and $f_i = \sigma(m_i - s_i)$. Define*

$$h_i = \int_0^1 \sigma'(m_i - ts_i)\,dt \in (0, 1/4], \qquad \zeta_i = \frac{w_i - 1}{w_i^* - 1} \in [0,1]$$

when $w_i^* \neq 1$. When $w_i^* = 1$, set $\zeta_i = 1$ for $\kappa > 1$ and $\zeta_i = 0$ for $\kappa = 1$. Then the actual update satisfies

$$G_{\mathcal{P}} - G_{\mathrm{S}} = -\frac{1}{n}\sum_i \zeta_i h_i(\lambda\Delta_i - D_i^{\mathcal{P}})b_i. \tag{19}$$

Proof. The clip interval contains one, so it contracts the signed difference $w_i^* - 1$ without changing its sign; this proves the range for ζ_i. The fundamental theorem of calculus gives $p_i - f_i = h_i s_i$, including $s_i = 0$. Since $(w_i^* - 1)(1 - p_i) = p_i - f_i$, $(w_i - 1)g_i = -\zeta_i h_i s_i b_i$. Average over tokens. □

This identity gives the effective historical displacement of each nonlinear margin at any checkpoint, including with active clipping. To relate it to parameter displacement, let $u = \theta - \theta_0$. If ∇m_i is M_i-Lipschitz along the segment between the endpoints, then

$$|\Delta_i - b_i^\top u| \le \tfrac{1}{2} M_i \|u\|^2.$$

Define the positive-semidefinite matrix and competitor vector

$$C_\theta = \frac{1}{n} \sum_i \zeta_i h_i b_i b_i^\top, \qquad B_\theta = \frac{1}{n} \sum_i \zeta_i h_i D_i^{\mathcal{P}} b_i.$$

Equation (19) implies the exact decomposition

$$G_{\mathrm{S}} - G_{\mathcal{P}} = \lambda C_\theta u - B_\theta + e_\theta, \qquad \|e_\theta\| \le \frac{\lambda \|u\|^2}{2n} \sum_i \zeta_i h_i M_i \|b_i\|. \tag{20}$$

The remainder quantifies nonlinear margin curvature. For $v = \nabla R(\theta)$ the risk coefficient obeys

$$\mathcal{A}_{\mathcal{P}} \ge \lambda \langle -v, C_\theta u \rangle + \langle v, B_\theta \rangle - \|v\| \|e_\theta\|.$$

This gives a sufficient condition linking aligned parameter history to the multiclass risk comparison through the competitor term and remainder.

Proposition B.1 (Local feedback for smooth multiclass models)**.** *For a finite set of fixed states and partitions, positive reference probabilities, twice continuously differentiable logits near θ_0, and $\kappa > 1$, the weighted field satisfies, near the reference,*

$$G_{\mathcal{P}}(\theta_0 + u) = G_{\mathrm{S}}(\theta_0 + u) - \lambda C_0 u + O(\|u\|^2), \quad C_0 = \frac{1}{n} \sum_i \frac{\nabla p_i(\theta_0) \nabla p_i(\theta_0)^\top}{q_i(1 - q_i)} \succeq 0. \tag{21}$$

If $J_{i0} = \partial z_i / \partial \theta|_{\theta_0}$ and $Q_i = P_i(\theta_0)$, define

$$F_0 = \frac{1}{n} \sum_i J_{i0}^\top [\mathrm{diag}(Q_i) - Q_i Q_i^\top] J_{i0}.$$

Then $0 \preceq C_0 \preceq F_0$. The first-order correction is independent of the fixed partition.

Proof. At θ_0 the reference equals the current distribution and $w_i = 1$. Since $\kappa > 1$ and there are finitely many states, continuity provides a common neighborhood with inactive clipping. For the geometric normalizer $Z_i = \sum_C P_i(\theta)(C)^{1-\lambda} P_i(\theta_0)(C)^\lambda$, any directional derivative at θ_0 satisfies

$$DZ_i = (1 - \lambda) \sum_C DP_i(\theta)(C) = 0, \qquad Df_i = (1 - \lambda) Dp_i.$$

Differentiating the raw residual ratio at θ_0 gives

$$\nabla w_i(\theta_0) = \frac{\lambda \nabla p_i(\theta_0)}{1 - q_i}.$$

Multiplying its first-order change by $g_i(\theta_0) = -\nabla p_i(\theta_0)/q_i$ and averaging proves Eq. (21), with a locally bounded second-order remainder.

For the matrix bound, fix a parameter direction v, let $a = J_{i0} v$, and let $\bar{a} = \sum_k Q_{i,k} a_k$. The per-token quadratic form is

$$v^\top C_{i0} v = \frac{q_i}{1 - q_i} (a_{y_i} - \bar{a})^2.$$

The corresponding form for F_{i0} is $\mathrm{Var}_{k \sim Q_i}(a_k)$. Splitting this variance into the target and its complement gives

$$\mathrm{Var}_{Q_i}(a_k) = \frac{q_i}{1 - q_i} (a_{y_i} - \bar{a})^2 + (1 - q_i) \, \mathrm{Var}_{Q_i(\cdot | k \ne y_i)}(a_k).$$

Hence $C_{i0} \preceq F_{i0}$, and averaging proves the claim. In binary affine logistic models the conditional variance vanishes and $C_0 = F_0 = \nabla^2 L(\theta_0)$. For general nonlinear models the full Hessian additionally contains logit-curvature terms. □

Where partition differences first appear. For three-times continuously differentiable probabilities, let $U_i = \partial u_{\theta,i}/\partial\theta|_{\theta_0}$ be the Jacobian of the conditional competitor distribution on a fixed partition. Taylor expansion, using that its coordinates sum to one, gives

$$D_i^{\mathcal{P}}(\theta_0 + u) = \frac{\lambda(1-\lambda)}{2} u^\top U_i^\top \operatorname{diag}(1/u_{0,i}) U_i u + O(\|u\|^3). \tag{22}$$

Indeed, for $r = u_{0,i}$ and $e = u_{\theta,i} - r$ with $\sum_C e_C = 0$, $\sum_C r_C^\lambda (r_C + e_C)^{1-\lambda} = 1 - \lambda(1-\lambda)\sum_C e_C^2/(2r_C) + O(\|e\|^3)$; taking $-\log$ proves the display. The binary conditional distribution is one-dimensional and constant, so this term vanishes. Partition effects therefore first appear at second order around the chosen reference.

B.4 Finite-step task-risk comparison and ablation identities

Proof of Theorem 3.2. Assume that ∇R is L-Lipschitz on a convex region containing θ, θ_{S}^+, and $\theta_{\mathcal{P}}^+$. By integration along the segment joining the two updated parameters,

$$R(\theta_{\mathrm{S}}^+) - R(\theta_{\mathcal{P}}^+) = \eta \int_0^1 \langle \nabla R(\theta - \eta G_{\mathrm{S}} - t\eta\delta G), \delta G\rangle \, dt.$$

Subtract $\eta\langle \nabla R(\theta), \delta G\rangle$ and apply Lipschitz continuity and the triangle inequality to obtain the stronger two-sided estimate

$$\left| R(\theta_{\mathrm{S}}^+) - R(\theta_{\mathcal{P}}^+) - \eta\mathcal{A}_{\mathcal{P}} \right| \le L\eta^2 \|\delta G\|(\|G_{\mathrm{S}}\| + \tfrac{1}{2}\|\delta G\|). \tag{23}$$

Theorem 3.1 and stop-gradient give $\langle \nabla R(\theta), \delta G\rangle = n^{-1}\sum_i (w_i^{\mathcal{P}} - 1)a_i = \mathcal{A}_{\mathcal{P}}$. The lower bound and the stated sufficient step-size condition follow. The clipped weights are detached throughout. If $\mathcal{B} = 0$ and $\mathcal{A} > 0$, every positive step whose connecting segments remain in the stipulated region yields a positive gap. □

Separating scale from assignment. For $\bar{w} = n^{-1}\sum_i w_i$ and $\bar{a} = n^{-1}\sum_i a_i$, define $\mathrm{Cov}_n(w,a) = n^{-1}\sum_i (w_i - \bar{w})(a_i - \bar{a})$. Then

$$\mathcal{A}_{\mathcal{P}} = (\bar{w} - 1)\bar{a} + \mathrm{Cov}_n(w,a), \qquad \mathcal{A}_{\mathcal{P}} - \mathcal{A}_{\mathrm{bin}} = -\frac{1}{n}\sum_i (w_i^{\mathrm{bin}} - w_i^{\mathcal{P}})a_i. \tag{24}$$

Here $\mathcal{A}_{\mathrm{bin}}$ is the same risk coefficient for the binary partition. Replacing all weights by $\bar{w}$ removes exactly the covariance from this first-order risk coefficient. Uniform random permutation of the weights has the same effect in expectation, conditional on the batch. For fixed confidence bins I_b of size n_b, shuffling only within bins instead gives

$$\mathcal{A}_{\mathrm{Full}} - \mathbb{E}_{\mathrm{shuffle}}\mathcal{A}_{\mathrm{Shuffle}} = \sum_b \frac{n_b}{n} \mathrm{Cov}_{I_b}(w,a). \tag{25}$$

Equation (25) isolates the first-order covariance within bins; finite-step and optimizer effects enter separately. A width-0.1 bin groups a range of target probabilities.

When competitor refinement helps. Let $r_i = w_i^{\mathrm{bin}} - w_i^{\mathcal{P}} \ge 0$. Then $G_{\mathcal{P}} - G_{\mathrm{bin}} = -n^{-1}\sum_i r_i g_i$, proving the second identity in Eq. (24). A direct application of the risk comparison, now taking Binary as the baseline, yields

$$R(\theta_{\mathrm{bin}}^+) - R(\theta_{\mathcal{P}}^+) \ge -\frac{\eta}{n}\sum_i r_i a_i - L\eta^2 \|G_{\mathcal{P}} - G_{\mathrm{bin}}\|(\|G_{\mathrm{bin}}\| + \tfrac{1}{2}\|G_{\mathcal{P}} - G_{\mathrm{bin}}\|). \tag{26}$$

Equation (26) gives a finite-step sufficient condition for improving risk by attenuating negatively aligned contributions. Attenuating positively aligned contributions can instead increase risk.

Stochastic batches and practical optimizers. These identities compare alternative updates from the same checkpoint and realized batch. With integrability and a common smoothness region, expectations can be taken in Eq. (23); the remainder involves $\mathbb{E}[\|\delta G\|(\|G_{\mathrm{S}}\| + \|\delta G\|/2)]$. For arbitrary optimizer-produced parameter displacements d_{S} and $d_{\mathcal{P}}$, let $e = d_{\mathcal{P}} - d_{\mathrm{S}}$. The exact same argument gives

$$|R(\theta + d_{\mathrm{S}}) - R(\theta + d_{\mathcal{P}}) + \langle \nabla R(\theta), e\rangle| \le L\|e\|(\|d_{\mathrm{S}}\| + \tfrac{1}{2}\|e\|). \tag{27}$$

For AdamW, momentum, or global gradient clipping, e depends on the initial optimizer state and the actual parameter displacements. A differentiable validation loss is one choice of risk in this comparison; the endpoint ablations report benchmark accuracy.

B.5 TOP-K PLUS TAIL: QUANTITATIVE WEIGHT, UPDATE, AND RISK ERROR

Let H_i contain the retained non-target singleton tokens, and let T_i be the remaining tail. The full-vocabulary comparator keeps the same λ, κ, losses, and current/reference predictions and resolves every tail token as a singleton. Define the unresolved tail's share of competitor overlap and the associated gradient-error budget by

$$t_i^{(K)} = \frac{P_i(\theta_0)(T_i)^\lambda P_i(\theta)(T_i)^{1-\lambda}}{\sum_{C\neq\{y_i\}} P_i(\theta_0)(C)^\lambda P_i(\theta)(C)^{1-\lambda}}, \qquad \varepsilon_K = \frac{1}{n}\sum_i w_i^{(K)} t_i^{(K)} \|g_i\|. \tag{28}$$

For an empty tail, set $t_i^{(K)} = 0$.

Theorem B.1 (Error due to an unresolved tail)**.** *For $0 < \lambda < 1$,*

$$0 \le w_i^{(K)} - w_i^{\text{full}} \le w_i^{(K)} t_i^{(K)} \le \kappa t_i^{(K)}, \qquad \|G_K - G_{\text{full}}\| \le \varepsilon_K. \tag{29}$$

If R has L-Lipschitz gradient on the segment joining the two updated parameters, then

$$|R(\theta_K^+) - R(\theta_{\text{full}}^+)| \le \eta \|\nabla R(\theta_K^+)\| \varepsilon_K + \tfrac{1}{2} L \eta^2 \varepsilon_K^2. \tag{30}$$

In particular,

$$R(\theta_{\text{S}}^+) - R(\theta_K^+) \ge R(\theta_{\text{S}}^+) - R(\theta_{\text{full}}^+) - \eta \|\nabla R(\theta_K^+)\| \varepsilon_K - \tfrac{1}{2} L \eta^2 \varepsilon_K^2. \tag{31}$$

At $\lambda = 0$ or 1, or $\kappa = 1$, the two updates coincide exactly.

Proof. Suppress i. Write

$$B_K = \sum_{v\in H} P_0(v)^\lambda P_\theta(v)^{1-\lambda} + b_T, \qquad b_T = P_0(T)^\lambda P_\theta(T)^{1-\lambda}.$$

Within a nonempty tail define $c_T = \sum_{v\in T} P_0(v \mid T)^\lambda P_\theta(v \mid T)^{1-\lambda} \in (0, 1]$. The full-vocabulary non-target mass is exactly

$$B_{\text{full}} = B_K - b_T(1 - c_T), \qquad \frac{c_{\text{full}}}{c_K} = \frac{B_{\text{full}}}{B_K} = 1 - t^{(K)}(1 - c_T).$$

By the refinement inequality, $w_{\text{full}} \ge (c_{\text{full}}/c_K) w_K$. Therefore the sharper inequality

$$0 \le w_K - w_{\text{full}} \le w_K t^{(K)}(1 - c_T) \le w_K t^{(K)}$$

holds even with active clipping. Multiplying by $\|g_i\|$, averaging, and using the triangle inequality proves the update bound. The parameter updates differ by at most $\eta\varepsilon_K$. Taylor's inequality centered at θ_K^+ gives

$$|R(\theta_{\text{full}}^+) - R(\theta_K^+)| \le \|\nabla R(\theta_K^+)\| \, \|\theta_{\text{full}}^+ - \theta_K^+\| + \tfrac{1}{2} L \|\theta_{\text{full}}^+ - \theta_K^+\|^2,$$

which proves the risk and transfer bounds. The endpoint cases follow from partition-independence of the reference, or from unit clipping. □

Interpreting the top-K bound. Reference head/tail masses and current cell probabilities determine $t_i^{(K)}$. Resolving within-tail probabilities gives the sharper factor $1 - c_T$. The conservative bound depends on both tail mass and competitor overlap. Refining nested heads moves the weight toward its full-vocabulary value.

Conditional multi-step extension. For a prescribed common sequence of batches, let $G_{\text{full},t}$ be L_t-Lipschitz on a region containing both trajectories, and assume $\|G_{K,t}(x) - G_{\text{full},t}(x)\| \le \epsilon_t$ uniformly on that region. Plain gradient descent from identical parameters satisfies

$$\|\theta_{K,T} - \theta_{\text{full},T}\| \le \sum_{s=0}^{T-1} \eta_s \epsilon_s \prod_{j=s+1}^{T-1} (1 + \eta_j L_j).$$

This follows by induction from $d_{t+1} \le (1 + \eta_t L_t) d_t + \eta_t \epsilon_t$. A risk with gradient norm bounded by B on the connecting segment then differs by at most B times this bound. This stability bound applies to plain gradient descent with the stipulated common batches and uniform error bounds; adaptive optimizers and changing rollout laws require separate recursions.

B.6 THE BINARY-AFFINE SPECIAL CASE, FOR GENERAL MIXING STRENGTH

For observation i, absorb its binary label into an affine signed margin $m_i(\theta)$, and write $p_i = \sigma(m_i(\theta))$ and $q_i = \sigma(m_i(\theta_0))$. Without active clipping,

$$f_i = \sigma((1-\lambda)m_i(\theta) + \lambda m_i(\theta_0)), \qquad w_i^* = (1-f_i)/(1-p_i).$$

Let $\vartheta_\lambda = \theta_0 + (1-\lambda)(\theta - \theta_0)$. Affineness implies

$$w_i^* \nabla_\theta \ell_i(\theta) = -(1-f_i)\nabla m_i = \nabla_\vartheta \ell_i(\vartheta)|_{\vartheta=\vartheta_\lambda}. \tag{32}$$

Averaging gives

$$G_{\mathrm{R}}(\theta) = G_{\mathrm{S}}(\vartheta_\lambda). \tag{33}$$

For $\lambda = 1/2$ this is the previous midpoint identity. The right side evaluates the ordinary gradient at ϑ_λ; differentiating through ϑ_λ would introduce an extra factor $1-\lambda$. The identity applies to binary affine logistic models; Theorem 3.1 and Lemma B.1 characterize the nonlinear multiclass field.

B.7 A CONTROLLED EXPECTED-RISK ADVANTAGE

In the separately specified quadratic model of Eq. (1), the continuation field corresponding to the preceding evaluation identity is $G_{\mathrm{R}}^{\mathrm{quad}}(u) = H((1-\lambda)u - \mu - \xi)$. The model's warm-up trajectory begins at the selected reference, $u(0) = 0$, so the resulting risk comparison covers this run-start reference. From the shared SFT checkpoint $u(\tau)$, compare

$$\begin{aligned} u_{\mathrm{S}}^+ &= u(\tau) - \eta H(u(\tau) - \mu - \xi), \\ u_{\mathrm{R}}^+ &= u(\tau) - \eta H((1-\lambda)u(\tau) - \mu - \xi) = u_{\mathrm{S}}^+ + \eta\lambda H u(\tau). \end{aligned} \tag{34}$$

Theorem B.2 (Controlled quadratic continuation)**.** *For $0 < \tau < \infty$, $0 < \lambda \le 1$, and the shared empirical fluctuation ξ used in the warm-up and both compared updates,*

$$\mathbb{E}R_{\mathrm{quad}}(u_{\mathrm{S}}^+) - \mathbb{E}R_{\mathrm{quad}}(u_{\mathrm{R}}^+) = \eta\Gamma_{\tau,\lambda} - \eta^2 B_{\tau,\lambda}, \tag{35}$$

where

$$\begin{aligned} \Gamma_{\tau,\lambda} &= \lambda \sum_j \gamma_j^2 A_j(\tau), \\ B_{\tau,\lambda} &= \frac{\lambda}{2} \sum_j \gamma_j^3 \rho_j [2(1-\rho_j) + \lambda\rho_j](\mu_j^2 + v_j) \ge 0. \end{aligned} \tag{36}$$

If $\Gamma_{\tau,\lambda} > 0$, then $B_{\tau,\lambda} > 0$ and $0 < \eta < \Gamma_{\tau,\lambda}/B_{\tau,\lambda}$ gives a strictly positive gap. At $\lambda = 0$, the two updates coincide.

Proof. Let

$$s_j = \rho_j + \eta\gamma_j(1-\rho_j), \qquad r_j = s_j + \eta\gamma_j\lambda\rho_j.$$

The two updated coordinates are $s_j(\mu_j + \xi_j)$ and $r_j(\mu_j + \xi_j)$. Since $\mathbb{E}[c(\mu_j + \xi_j) - \mu_j]^2 = (1-c)^2\mu_j^2 + c^2 v_j$,

$$\mathbb{E}R_{\mathrm{quad}}(u_{\mathrm{S}}^+) - \mathbb{E}R_{\mathrm{quad}}(u_{\mathrm{R}}^+) = \frac{1}{2}\sum_j \gamma_j(r_j - s_j)[(2 - r_j - s_j)\mu_j^2 - (r_j + s_j)v_j].$$

Substituting $r_j - s_j = \eta\gamma_j\lambda\rho_j$ and $r_j + s_j = 2\rho_j + \eta\gamma_j[2(1-\rho_j) + \lambda\rho_j]$ yields the stated coefficients. Nonnegativity and the step-size condition follow because $0 < \rho_j < 1$. □

Along the warm-up trajectory, the coordinate fields obey

$$G_{\mathrm{R},j}^{\mathrm{quad}}(u(\tau)) = \frac{1-(1-\lambda)\rho_j}{1-\rho_j} G_{\mathrm{S},j}^{\mathrm{quad}}(u(\tau)). \tag{37}$$

If all ρ_j coincide, the step is a uniform rescaling; otherwise it is direction-dependent. The ratio grows when $\rho_j \to 1$ partly because the SFT denominator vanishes, while the extra update $\eta\lambda H u$ stays finite for fixed u. As $A_j \to -v_j$, reinforcing nearly exhausted noisy directions can increase risk. The multiclass risk condition and refinement effects are given in Eqs. (7) and (26).

C OPD GRADIENT AND TEACHER-CONSISTENCY PROOFS

C.1 THE SAMPLED REVERSE-KL GRADIENT

Fix a prefix h and let $p_i = \pi_\theta(i \mid h)$ and $\nu_i = \pi_T(i \mid h)$ be strictly positive student and frozen teacher probabilities over a finite vocabulary $\mathcal{V}$. The reverse-KL objective is

$$\mathcal{D}_h(\theta) = \mathrm{KL}(p\|\nu) = \sum_{i\in\mathcal{V}} p_i \log \frac{p_i}{\nu_i}. \tag{38}$$

Since $\sum_i \nabla_\theta p_i = 0$, differentiation gives

$$\begin{aligned}\nabla_\theta \mathcal{D}_h &= \sum_i \left(\log \frac{p_i}{\nu_i} + 1\right) \nabla_\theta p_i \\ &= \sum_i p_i \log \frac{p_i}{\nu_i} \nabla_\theta \log p_i \\ &= \mathbb{E}_{a\sim p}\left[\log \frac{p_a}{\nu_a} \nabla_\theta \log p_a\right].\end{aligned} \tag{39}$$

For a fixed sampled token a, the k2 loss in Eq. (8) has gradient $\nabla_\theta[\frac{1}{2}\log^2(p_a/\nu_a)] = \log(p_a/\nu_a)\nabla_\theta \log p_a$. Its expectation therefore equals the fixed-prefix reverse-KL gradient when a is sampled from the same full student distribution p. Prefixes and sampled tokens are held fixed during differentiation; the expectation is taken afterwards. The expected value of the k2 loss differs from $\mathrm{KL}(p\|\nu)$.

C.2 A TEACHER-CONSISTENT POTENTIAL

Fix h and let $p_i = \pi_\theta(i \mid h)$, $q_i = \pi_0(i \mid h)$, and $\nu_i = \pi_T(i \mid h)$ be strictly positive student, historical-reference and teacher distributions over the finite vocabulary. The historical reference and teacher are frozen. Assume the rollout distribution equals the current student distribution, and sample $a \sim p$. Let $d_a = \log(p_a/\nu_a)$ and $\ell_a^{\mathrm{OPD}} = \frac{1}{2}d_a^2$. For a fixed rollback strength $\lambda \in [0,1]$, the binary-partition weight is

$$\omega(u,v) = \mathrm{clip}\left(\frac{1-\sigma((1-\lambda)\,\mathrm{logit}(u) + \lambda\,\mathrm{logit}(v))}{1-u}, \kappa^{-1}, \kappa\right).$$

OPD training uses the clipped geometric weight on the pre-update student's top-15 partition (Appendix A.1). That weight is detached and takes values in $[\kappa^{-1}, \kappa]$. Assume policy matching as specified in Appendix A.1: the stored student probability used for weighting equals the current student probability in the differentiable loss.

We construct a potential for the expected update and prove its gradient, minimum, and stationarity properties.

Theorem C.1 (Fixed-prefix distillation potential)**.** *Define*

$$\Phi(p;q,\nu) = \sum_i \int_{\nu_i}^{p_i} \omega(u,q_i) \log \frac{u}{\nu_i}\,\mathrm{d}u. \tag{40}$$

Then the expected actual sampled gradient satisfies

$$\mathbb{E}_{a\sim p}\left[\nabla_z \left\{\mathrm{sg}(\omega(p_a,q_a))\tfrac{1}{2}\log^2(p_a/\nu_a)\right\}\right] = \nabla_z \Phi(p;q,\nu). \tag{41}$$

Furthermore,

$$\kappa^{-1}\mathrm{KL}(p\|\nu) \le \Phi(p;q,\nu) \le \kappa \mathrm{KL}(p\|\nu). \tag{42}$$

Thus $\Phi \ge 0$, with equality iff $p = \nu$; this is also its unique interior stationary distribution in logit coordinates. Logits themselves are determined only up to an additive constant. For $\omega \equiv 1$, $\Phi = \mathrm{KL}(p\|\nu)$.

Proof. **Gradient representation.** For a fixed sampled token, the teacher and the weight are detached:

$$\nabla_z[\mathrm{sg}(\omega(p_a,q_a))\ell_a^{\mathrm{OPD}}] = \omega(p_a,q_a)\log(p_a/\nu_a)\nabla_z \log p_a.$$

Taking the expectation over $a \sim p$ gives

$$\sum_i p_i \omega(p_i, q_i) \log(p_i/\nu_i) \nabla_z \log p_i = \sum_i \omega(p_i, q_i) \log(p_i/\nu_i) \nabla_z p_i.$$

The clipped weight is continuous, including at clipping boundaries. The fundamental theorem of calculus gives $\partial\Phi/\partial p_i = \omega(p_i, q_i) \log(p_i/\nu_i)$; the chain rule therefore proves (41). Differentiation acts on each sampled term, and the expectation is taken afterwards.

KL bounds. For the bounds, put

$$H_i = \int_{\nu_i}^{p_i} \log(u/\nu_i)\, \mathrm{d}u = p_i \log(p_i/\nu_i) - p_i + \nu_i \geq 0.$$

When $p_i \geq \nu_i$, the integrand is nonnegative. When $p_i < \nu_i$, reversing integration limits gives $H_i = \int_{p_i}^{\nu_i} \log(\nu_i/u)\, \mathrm{d}u$, again with a nonnegative integrand. In either orientation,

$$\kappa^{-1} H_i \leq \int_{\nu_i}^{p_i} \omega(u, q_i) \log(u/\nu_i)\, \mathrm{d}u \leq \kappa H_i.$$

Summing and using $\sum_i p_i = \sum_i \nu_i = 1$ proves the KL bounds, the unique minimum at $p = \nu$, and the uniform-weight identity.

Stationary distributions. For stationarity, write $b_i = \omega(p_i, q_i) \log(p_i/\nu_i)$ and $\bar{b} = \sum_j p_j b_j$. The softmax Jacobian yields $\partial\Phi/\partial z_i = p_i(b_i - \bar{b})$. With positive p_i, a stationary point requires all b_i to agree. If $p \neq \nu$, normalization implies some $p_i > \nu_i$ and some $p_j < \nu_j$. Positive weights then give $b_i > 0 > b_j$, a contradiction. At $p = \nu$, all log-ratios are zero. □

The historical guide signal enters Φ through ω, so predictive change from the selected frozen checkpoint shapes the fixed-prefix update field. The potential and bounds also hold for any continuous, positive, uniformly bounded separable weight $\omega(u, q_i)$ with the corresponding bounds. Thus this is a target-preservation property, not a REWIND-specific performance advantage. The unique interior stationary *distribution* is proved in free logit coordinates; arbitrary shared neural parameters can introduce additional stationary points through their Jacobians. The derivation fixes one prefix, assumes policy matching, and does not establish convergence over evolving rollout distributions.

D DISCUSSION

To our knowledge, REWIND is the first method to use a shared historical guide signal rule in both SFT and OPD; the compared allocation methods each target only one of these settings. It rescales each existing token loss using predictions from a frozen earlier checkpoint of the same run, adds no trainable parameters, and leaves the data, teacher, optimizer, and training loop unchanged; the extra cost is one frozen-model scoring pass. Because the guide signals are positive, the demonstrations and the teacher still determine each term's correction direction. Without additional training data or any model stronger than those the base objective already uses, REWIND substantially improved the domain average over standard SFT or uniform OPD in every evaluated setting. The main advantage of REWIND is that it is simple and effective.

These findings have limitations. Large language model training remains largely opaque: how a token-level reweighting propagates through shared parameters to downstream behavior cannot be fully traced. Our theory characterizes the clipped update and gives conditions for one-step risk improvement, but it does not explain why REWIND improves long-run performance. We therefore rely on empirical tests: matched ablations of token assignment, competitor information, historical fusion, reference checkpoint, and clipping; controls for longer training and a higher learning rate; repeated training seeds; and token-level analysis of prediction changes. These experiments are consistent with the proposed mechanism but do not establish a complete causal account. In addition, computational constraints limited our evaluation to models of at most 4B parameters, so whether the gains persist at larger scales remains to be verified. The evaluation covers two post-training settings (SFT and OPD), two model scales, and four domains: mathematics, code, logical reasoning, and instruction following. Within these limits, REWIND offers a simple yet effective way to use training history in post-training.