

CPSD: Counterfactual Patches Self-Distillation in RLVR

Yixiao Zhou^{1,2}, Dongzhou Cheng^{2,3}, Hehe Fan¹, Yu Cheng^{2,4}

¹Zhejiang University, ²Shanghai Innovation Institute,
³Southeast University, ⁴The Chinese University of Hong Kong
{12421181, hehefan}@zju.edu.cn, 230249457@seu.edu.cn,
chengyu@cse.cuhk.edu.hk

Correspondence: chengyu@cse.cuhk.edu.hk, hehefan@zju.edu.cn

Abstract

Self-distillation in reinforcement learning with verifiable rewards (RLVR) uses successful in-group rollouts to teach the failed ones, with no external teacher needed. Whether it actually helps turns on three closely related design questions that the canonical method does not handle carefully: how to construct an appropriate teacher context for a given loser, where in the response the distillation loss should act, and what kind of signal it should carry. The canonical Self-Distillation Policy Optimization (SDPO) instantiation answers all three by default: a random in-group winner, a global distillation loss on the whole response, and a Jensen–Shannon divergence on the full top- K distribution. These three defaults compound into a catastrophic late-training failure that short-horizon evaluation hides. We propose CPSD (Counterfactual Patches Self-Distillation), which replaces each default in turn: pick the teacher by minimum embedding-weighted edit distance, localise distillation to an adaptive-stop counterfactual patch beginning at the first divergent token, and replace JS with a Plackett–Luce listwise loss that preserves only the teacher’s ranking. Across six reasoning benchmarks at a fixed 150-step budget, CPSD eliminates the SDPO cliff and lifts average accuracy by +5.7 pp over GRPO on Qwen3-8B and +4.2 pp on Qwen3-4B.

1 Introduction

Reinforcement learning with verifiable rewards (RLVR) (Shao et al., 2024; Guo et al., 2025a; Yu et al., 2026a; Team et al., 2025; Hao et al., 2026a; Wang et al., 2026b) has become the default recipe for reasoning-LLM post-training. A natural augmentation, when several rollouts of a single prompt happen to land on both correct and incorrect answers, is to let the correct ones teach the incorrect ones in the same training step, a form of *in-group self-distillation* (Hinton et al., 2015). The appeal

is direct: no external reward model is needed, no extra inference is required, and the teacher and student share the same architecture and decoding distribution by construction (Agarwal et al., 2024; Li et al., 2026b). The canonical instantiation, Self-Distillation Policy Optimization (SDPO) (Hübotter et al., 2026), reports gains on tool-use and reasoning benchmarks.

Looking at how SDPO actually behaves over training, however, we find that whether in-group self-distillation works inside a policy-optimization loop turns on three closely related design questions that the canonical method does not handle carefully: (Q1) how to construct an appropriate teacher context for a given loser; (Q2) which positions inside the loser’s response the distillation loss should act on; and (Q3) what form of signal the loss should carry. SDPO answers all three by default: a uniformly random in-group winner, a global loss across the loser’s entire response, and a Jensen–Shannon divergence on the full top- K distribution (Agarwal et al., 2024). All three defaults are wrong, in ways that compound rather than cancel. A random teacher steers the loser onto an unrelated reasoning trajectory; a global loss penalises tokens the loser had already gotten right; and JS forces the student to mirror the teacher’s full distribution shape, collapsing the multi-mode reasoning capacity that RLVR is meant to preserve. The three failures compound into a catastrophic late-training cliff (Figure 1a) that short-horizon evaluation hides.

We propose CPSD (Counterfactual Patches Self-Distillation), which replaces each default with a small, composable change. For Q1, we choose the teacher by minimum embedding-weighted edit distance (“SEM”, for Semantic Edit Matching): the in-group winner whose semantic trajectory is closest to the loser’s in the model’s own representation space. For Q2, we localise the distillation loss to a short adaptive-stop window beginning at the first position where the loser and the chosen teacher

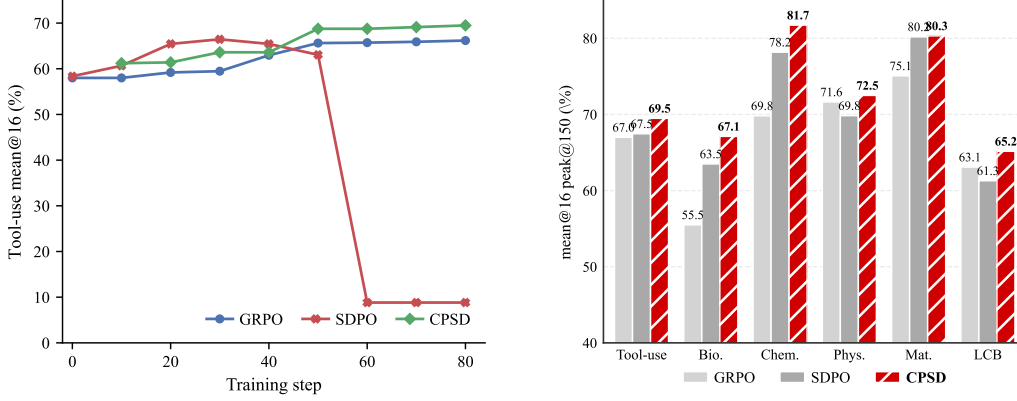


Figure 1: **The SDPO cliff and CPSD’s early-budget gain.** (left) Tool-use mean@16: SDPO peaks early then collapses; CPSD-LM clears both baselines without collapsing. (right) Per-task peak@150 on Qwen3-8B; CPSD-LM attains the highest peak on every task (values in Table 1).

diverge, a *counterfactual patch* that supplies gradient at exactly the decision point and nowhere else. For Q3, we replace JS inside the patch with a Plackett–Luce listwise loss (Xia et al., 2008) that preserves only the teacher’s ranking of the top- K candidates; the student inherits the teacher’s preference order while keeping its own temperature and calibration intact. Across six reasoning benchmarks at a fixed 150-step compute budget, CPSD eliminates the SDPO cliff and lifts average accuracy by +5.7 pp over GRPO on Qwen3-8B and +4.2 pp on Qwen3-4B.

Our contributions are as follows:

- A re-framing of in-group RLVR self-distillation around three closely related design questions (teacher context, distillation scope, and signal form), and the observation that the SDPO defaults are wrong on all three.
- CPSD, a composable instantiation that addresses each question independently: SEM teacher selection, first-divergence adaptive-stop counterfactual patch, and Plackett–Luce listwise loss inside the patch.
- Empirical evidence on six reasoning benchmarks and two backbones (Qwen3-8B, Qwen3-4B) that CPSD removes the SDPO late-training cliff and lifts average accuracy over GRPO.

2 Background

Notation. For a prompt x we sample a group G of N rollouts $\{y_g\}_{g=1}^N$ with scalar verifiable rewards $\{r_g\}$. $\mathcal{W}_g = \{g : r_g > 0\}$ and $\mathcal{L}_g = \{g : r_g = 0\}$ partition the group into winners and losers.

A group is *mixed* if $|\mathcal{W}_g| \geq 1$ and $|\mathcal{L}_g| \geq 1$. The student policy is π_θ ; SDPO and CPSD also maintain an EMA teacher $\pi_{\theta'}$ with the same architecture. Per-token distributions are $\mathbf{p}_t$ (student) and $\mathbf{q}_t$ (teacher); the student log-prob of the token at teacher rank i is s_i .

GRPO. Group-Relative Policy Optimization (Shao et al., 2024) normalises rewards within a group of N rollouts and applies a clipped PPO-style update (Schulman et al., 2017). For rollout g with reward r_g , the advantage is shared across all tokens $t \in y_g$:

$$A_t = \frac{r_g - \mu_G}{\sigma_G}, \quad t \in y_g,$$

where μ_G and σ_G are the mean and standard deviation of $\{r_{g'}\}_{g'=1}^N$ within the group. The corresponding GRPO loss is then:

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{g,t} [\min(\rho_t A_t, \tilde{\rho}_t A_t)],$$

where $\rho_t = \pi_\theta(y_t | y_{<t}, x) / \pi_{\theta_{\text{old}}}(y_t | y_{<t}, x)$ is the importance ratio and $\tilde{\rho}_t = \text{clip}(\rho_t, 1-\epsilon, 1+\epsilon)$ is its clipped version.

SDPO baseline. Hübottter et al. (2026) augment GRPO with a global in-group distillation loss. For every loser ℓ in a mixed group, a “solution” is taken as `success_list[0]` (the first index in the per-group success list). Since this ordering reflects vLLM batch position, the picker is *effectively random*. The loss is Jensen–Shannon divergence between π_θ and $\pi_{\theta'}$ (the latter conditioned on the chosen winner’s response as “privileged context”) over the loser’s *entire* response, blended with GRPO via

a fixed coefficient. We inherit GRPO’s importance-sampling clip but replace the SDPO picker, loss window, and divergence form in turn (Section 4).

3 Where SDPO’s defaults break

We give the empirical evidence behind the three diagnoses, in the order posed in Section 1.

The late-training cliff is real and reproducible.

On the tool-use benchmark, paper-faithful SDPO reaches its training peak after roughly forty steps and loses almost everything within the next twenty: by step sixty the model can no longer reliably perform the task at all (Figure 1a). GRPO without distillation climbs more slowly but never collapses. CPSD climbs along the same early trajectory as SDPO and continues past it without the late drop; the same peak-end gap recurs across the six reasoning tasks (Tab. 1).

Teacher choice: the picker is effectively random.

Two diagnostic observations substantiate the claim that the SDPO picker is misspecified. (i) The codebase picks the first winner in batch order, with no semantic content. The induced picker is effectively

$$w_{\text{SDPO}}^* \sim \text{Unif}(\mathcal{W}_g),$$

in contrast to the embedding-distance picker $w^* = \arg \min_{w \in \mathcal{W}_g} \text{Lev}_E(y^\ell, y^w)$ introduced in Section 4.1. (ii) Against a token-prefix-longest baseline, the semantic picker selects a *different* winner on between roughly a quarter and two-fifths of all mixed-group pairs, depending on the task.

Distillation scope: global loss over-supervises.

The SDPO distillation loss is summed over the loser’s entire response,

$$\mathcal{L}_{\text{patch}}^{\text{SDPO}} = \sum_{t \in y^\ell} D_t(\mathbf{p}_t^\ell, \mathbf{q}_t^{w^*}),$$

with no preference for actual decision points. Two pieces of evidence suggest this scope is too wide. (i) Per-position student entropy $H(\mathbf{p}_t^\ell) := -\sum_v \mathbf{p}_t^\ell(v) \log \mathbf{p}_t^\ell(v)$ under paper-faithful SDPO on tool-use drops by an order of magnitude during the steps that precede the cliff, from roughly 0.21 to 0.03; the policy commits to a single solution path just before it loses the ability to recover from it. (ii) Per-batch divergence-geometry probes show that biology fires patches roughly five times more often per batch at six times the per-batch GRPO loss magnitude relative to tool-use.

The ratio-clip safety net of Lemma 1 guarantees $|c_{\text{eff}} \mathcal{L}_{\text{patch}}| \leq \kappa |\mathcal{L}_{\text{GRPO}}|$ but *scales with* $\mathcal{L}_{\text{GRPO}}$ and therefore cannot compensate for the cross-task spread in patch density.

Signal form: JS is too rigid for multi-mode reasoning.

The JS divergence at the heart of SDPO, $D_{\text{JS}}(\mathbf{p}_t^\ell \parallel \mathbf{q}_t^{w^*})$, matches the student and teacher on the full top- K distribution. On single-mode tasks like tool-use this is approximately the right constraint; on multi-mode reasoning tasks (biology, chemistry, physics, material) several valid solution paths coexist and the teacher’s distribution shape encodes one of them. JS asks the student to inherit *a particular distribution shape* when what it actually needs is the teacher’s *preference ordering* over candidates. A divergence that depends on the teacher only through its ranking σ_T — formalised in Proposition 3 — sidesteps this rigidity by construction. The factorial ablation in Table 2 confirms that swapping JS for such a ranking loss inside the patch contributes independently of the teacher-choice fix.

Independence of the three causes. The three defaults break along orthogonal axes, parameterising an in-group self-distillation method as the triple $(\pi, \mathcal{T}, D)$: a picker π (*which* gradient flows), a scope $\mathcal{T}$ (*where* it flows), and a divergence D (*what shape* it pushes toward). Theorem 4 shows that the gradient flow induced by the rank-preserving choice of D eliminates every teacher-order inversion in finite time, while Lemma 5 shows the localised scope leaves out-of-patch GRPO gradients undisturbed. A single-axis fix recovers only part of the late-training stability gap (Section 6); only when all three are corrected simultaneously does the cliff vanish and the peak-end gap close.

4 CPSD

CPSD keeps SDPO’s group structure and EMA teacher unchanged. The contribution is a redesign of the three components Section 3 identified as misspecified: the teacher picker, the distillation scope, and the divergence form. Figure 2 shows the pipeline; formal properties of each lever — patch-loss safety, ListMLE invariances, gradient-flow rank recovery, and out-of-patch GRPO preservation — are established in Appendix H.

4.1 Choosing the teacher: a semantic picker

For each loser $\ell \in \mathcal{L}_g$ in a mixed-reward group, we choose the winner with minimum embedding-

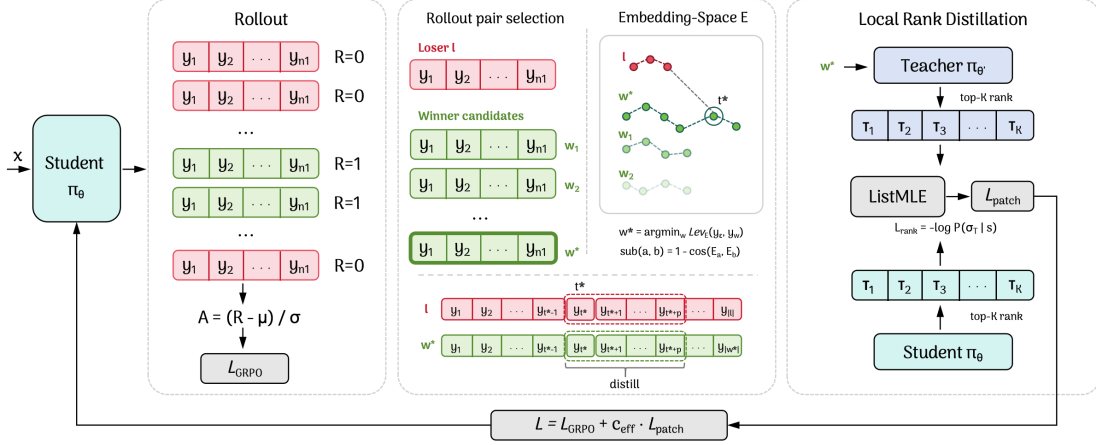


Figure 2: **CPSPD pipeline**. For each loser ℓ in a mixed-reward group, the SEM picker selects a winner w^* by embedding-weighted edit distance (not lexical prefix). The first-divergence position t^* defines an adaptive-stop patch window. Inside the window the student is supervised by a Plackett–Luce ListMLE rank loss against the EMA-teacher’s top- K order, capped by a ratio-clip safety net. The patch loss is *added* to GRPO, not replacing it.

weighted Levenshtein distance to ℓ :

$$w^* = \arg \min_{w \in \mathcal{W}_g} \text{Lev}_E(y^\ell, y^w), \quad (1)$$

where the substitution cost is the cosine distance between input embeddings drawn from the model’s own embedding matrix E : $\text{sub_cost}(a, b) = 1 - \cos(E_a, E_b)$. Insertion and deletion cost 1; the distance is capped at the first 128 tokens of each rollout, which keeps its cost negligible relative to a forward pass. We call this picker *SEM* for *Semantic Edit Matching*: substitution cost is driven by the model’s own embedding geometry rather than by raw token identity. Unlike a syntactic prefix-longest match, SEM selects the winner whose semantic trajectory is closest to the loser’s even when they paraphrase heavily from the first token.

4.2 Choosing the scope: a counterfactual patch

Once w^* is fixed, the first-divergence position $t^* = \min\{t : \arg \max \mathbf{p}_t^\ell \neq \arg \max \mathbf{p}_t^{w^*}\}$ identifies the earliest token at which the two trajectories disagree (if the two $\arg \max$ sequences never diverge no patch is fired for this loser). Empirically t^* lies early in the response across all five tasks (Fig. 5). The patch window T_{patch} starts at t^* and is adaptively truncated: distillation proceeds token-by-token until the student already agrees with the teacher locally — measured as per-token JS divergence between $\mathbf{p}_t^\ell$ and $\mathbf{p}_t^{w^*}$ below a small threshold ϵ for $k = 4$ consecutive positions — or a hard cap of 64 tokens is reached. A minimum length of 8 tokens prevents premature termination on transient

local agreement. We refer to this as the *adaptive-stop* window: the patch extends only as long as the student needs supervision to converge to the teacher’s local decision, and stops once it does. Tokens before t^* and past the adaptive cut-off receive no distillation gradient.

Aggregating across the loser set, the patch loss takes the form

$$\mathcal{L}_{\text{patch}} = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \frac{1}{|T_{\text{patch}}^{(\ell)}|} \sum_{t \in T_{\text{patch}}^{(\ell)}} D_t(\mathbf{p}_t^\ell \| \mathbf{q}_t^{w^*}),$$

where D_t is the in-patch divergence introduced in Section 4.3 below. Lemma 5 shows that positions outside $\bigcup_{\ell} T_{\text{patch}}^{(\ell)}$ receive no distillation gradient and propagate only the GRPO advantage signal.

The combination of SEM and patch localisation operationalises the *counterfactual* reading of in-group self-distillation: for each loser we ask “had the model made a different decision at the first divergence point, would a semantically similar trajectory have led to the correct answer?” Distillation pushes the student toward the positive instance of that counterfactual at exactly the location where it failed, and at no other location.

4.3 Choosing the signal: a ranking-only loss

Inside T_{patch} we replace JS divergence with a Plackett–Luce listwise loss (Xia et al., 2008). Let $\sigma_T \in S_K$ be the permutation that sorts the teacher’s top- K log-probabilities in descending order, and let s_i be the student log-probability of the token at teacher rank i . Under the Plackett–Luce model the

probability that the student’s scores reproduce the ranking σ_T factorises as

$$P_{\text{PL}}(\sigma_T \mid \mathbf{s}) = \prod_{i=1}^K \frac{\exp(s_i)}{\sum_{j=i}^K \exp(s_j)}. \quad (2)$$

The per-position negative log-likelihood, averaged over rank positions, is

$$\mathcal{D}_{\text{ListMLE}}(\mathbf{p}_t \parallel \mathbf{q}_t) = \frac{1}{K} \sum_{i=1}^K \left[\log \sum_{j=i}^K e^{s_j} - s_i \right], \quad (3)$$

computed in $O(K)$ memory with a reverse logcumsumexp pass over the student log-probs sorted in teacher order.

What distinguishes ListMLE from JS is its *asymmetric* treatment of teacher and student. The loss in Equation (3) depends on the teacher only through the permutation σ_T over the teacher’s top- K tokens: teacher logit magnitudes, temperature, and the residual mass on non-top- K tokens never enter. On the student side, the loss is invariant to a uniform additive shift of the student’s log-probabilities at a position (the same gauge freedom softmax already discards), so the student is supervised to reproduce the teacher’s *ordering* of those top- K tokens, not the absolute scale of its own logits. JS, by contrast, requires the student’s top- K distribution to match the teacher’s distribution shape pointwise. This is the explicit form of the “distribution shape versus ranking” distinction we draw in Section 3; Propositions 2–3 state these two invariances formally, and Theorem 4 shows that gradient flow under Equation (3) eliminates every teacher-order inversion in finite time.

4.4 Glue: slow EMA teacher and ratio-clip safety

After each optimiser step the teacher tracks the student via a slow exponential moving average, $\theta' \leftarrow (1 - \alpha)\theta' + \alpha\theta$, $\alpha = 0.01$. A frozen teacher ($\alpha = 0$, as recommended by Zhao et al. 2026d) loses roughly seven points on biology in our setting; a fast EMA ($\alpha \geq 0.05$) drives the late tool-use collapse described above. Slow EMA at 0.01 keeps the teacher useful without letting it lag.

To prevent intermittent magnitude spikes from dominating the additive loss, the effective patch coefficient is dynamically shrunk:

$$c_{\text{eff}} = c \cdot \min \left\{ 1, \frac{\kappa |\mathcal{L}_{\text{GRPO}}|}{|c \mathcal{L}_{\text{patch}}|} \right\}, \quad (4)$$

Algorithm 1 CPSD training step

Require: $\pi_\theta, \pi_{\theta'}$, prompt x , group size N , verifier $r(\cdot)$, ListMLE depth K , patch cap $L_{\text{max}}=64$, threshold ϵ , run-length $k=4$, EMA rate $\alpha_{\text{EMA}}=0.01$, ratio-clip $\kappa=0.25$, coef $c=0.25$

- 1: $\{y^g\}_{g=1}^N \sim \pi_\theta(\cdot \mid x)$, $r_g \leftarrow r(y^g)$
- 2: $\mathcal{W} \leftarrow \{g : r_g=1\}$, $\mathcal{L} \leftarrow \{g : r_g=0\}$
- 3: $\mathcal{L}_{\text{GRPO}} \leftarrow -\mathbb{E}_{g,t} [\min(\rho_t A_t, \text{clip}_\epsilon(\rho_t) A_t)]$
- 4: **if** $\mathcal{W} = \emptyset \vee \mathcal{L} = \emptyset$ **then**
- 5: $\mathcal{L}_{\text{patch}} \leftarrow 0$, $c_{\text{eff}} \leftarrow 0$ *unmixed group*
- 6: **else**
- 7: **for all** $\ell \in \mathcal{L}$ **do**
- 8: $w^* \leftarrow \arg \min_{w \in \mathcal{W}} \text{Lev}_E(y^\ell, y^w)$ *Q1*
- 9: $t^* \leftarrow \min \{t : \arg \max \mathbf{p}_t^\ell \neq \arg \max \mathbf{p}_t^{w^*}\}$ *Q2*
- 10: $T_{\text{patch}}^{(\ell)} \leftarrow \text{AdaptiveStop}(t^*; \epsilon, k, L_{\text{max}})$
- 11: **end for**
- 12: $\mathcal{L}_{\text{patch}} \leftarrow \frac{1}{|\mathcal{L}|} \sum_\ell \frac{1}{|T_{\text{patch}}^{(\ell)}|} \sum_{t \in T_{\text{patch}}^{(\ell)}} \mathcal{D}_{\text{ListMLE}}(\mathbf{p}_t^\ell \parallel \mathbf{q}_t^{w^*})$ *Q3*
- 13: $c_{\text{eff}} \leftarrow c \cdot \min \left\{ 1, \frac{\kappa |\mathcal{L}_{\text{GRPO}}|}{|c \mathcal{L}_{\text{patch}}|} \right\}$ *ratio-clip*
- 14: **end if**
- 15: $\theta \leftarrow \theta - \eta \nabla_\theta (\mathcal{L}_{\text{GRPO}} + c_{\text{eff}} \mathcal{L}_{\text{patch}})$
- 16: $\theta' \leftarrow (1 - \alpha_{\text{EMA}}) \theta' + \alpha_{\text{EMA}} \theta$ *teacher EMA*

with $\kappa = 0.25$ a fixed cap. Lemma 1 shows this clip guarantees $|c_{\text{eff}} \mathcal{L}_{\text{patch}}| \leq \kappa |\mathcal{L}_{\text{GRPO}}|$ whenever the patch is non-trivial. The clip almost never fires under JS but is load-bearing under ListMLE, whose logcumsumexp accumulation produces intrinsically larger raw magnitudes.

4.5 Algorithm

The per-microbatch objective is additive, $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{GRPO}} + c_{\text{eff}} \cdot \mathcal{L}_{\text{patch}}$, so GRPO continues to drive the global update even when no group is mixed. The variant CPSD-JS fixes only the first two questions (SEM picker plus patch localisation; JS divergence inside the patch) and serves as the Q1+Q2 ablation; the full CPSD-LM fixes all three. Comparing the two isolates the contribution of the ranking-only signal from the contribution of the counterfactual teacher choice and patch scope.

5 Experiments

Our experiments are organised around two questions that follow directly from the three-question framing of Section 3. First, does correcting all three SDPO defaults jointly recover the stability that paper-faithful SDPO loses? The main results below answer this. Second, is each of the three corrections necessary on its own? The factorial ablation in Section 6 answers this.

Table 1: **Main results.** Best mean@16 within the first 50 / 150 steps (fixed 150-step budget). Bold = best per column within each model block. *Average* is the unweighted mean over the six tasks.

	Chemistry		Physics		Biology		Materials		Tool Use		LCB		Average	
	@50	@150	@50	@150	@50	@150	@50	@150	@50	@150	@50	@150	@50	@150
Qwen3-8B														
+ GRPO	60.98	69.82	66.02	71.64	44.38	55.50	74.00	75.07	65.72	67.00	61.00	63.09	62.02	67.02
+ SDPO	72.95	78.18	67.03	69.84	54.75	63.50	73.54	80.19	67.46	67.46	57.00	61.29	65.45	70.08
+ CPSD	70.48	81.73	68.98	72.50	57.75	67.12	77.99	80.32	68.84	69.49	60.88	65.16	67.49	72.72
Qwen3-4B														
+ GRPO	63.66	70.33	63.44	71.25	36.75	50.25	75.27	74.73	59.74	61.58	56.60	61.48	59.24	64.94
+ SDPO	70.86	79.97	67.58	68.20	48.25	57.75	71.94	71.94	59.10	59.10	50.90	55.48	61.44	65.41
+ CPSD	68.42	78.81	65.62	72.27	55.00	58.63	77.53	79.92	61.58	62.13	57.80	63.20	64.33	69.16

5.1 Setup

Model and training stack. We run experiments on **Qwen3-8B** and **Qwen3-4B** (Yang et al., 2025); each backbone serves as both the on-policy student and the slow-EMA teacher. Training uses ver1 (Sheng et al., 2025), FlashAttention-2 (Dao et al., 2022), and vLLM for rollout (Kwon et al., 2023) on 4×H200. Each method runs to *exactly 150 training steps* on each backbone for an apples-to-apples comparison.

Tasks. Six reasoning benchmarks: **tool-use** (JSON exact-match verifier), the four **SciKnowEval** (Feng et al., 2024) domains **biology**, **chemistry**, **physics**, and **material** (MCQ verifier), and **LiveCodeBench-v6** (Jain et al., 2025) (code generation with sandbox-executed unit tests).

Metrics. We report validation mean@16 accuracy at two budgets: the peak within the first 50 training steps (*peak@50*, an early-training reading) and the peak within the full 150 steps (*peak@150*). The two columns together summarise both how quickly a method reaches a useful policy and how much additional budget is rewarded.

Baselines. **GRPO**, no distillation; and **SDPO**, paper-faithful with `solution_list[0]` picker and global JS loss. The main-table method is **CPSD-LM**. The Q1+Q2 ablation CPSD-JS, which fixes the teacher and scope but keeps JS inside the patch, appears in the factorial ablation (Tab. 2), not as a main-result row. All methods share the same optimisation hyper-parameters; full settings are listed in Appendix B.

5.2 Main results

Table 1 summarises peak@50 and peak@150 for CPSD-LM against the two baselines across all six tasks. Two patterns are worth highlighting.

The cliff is eliminated. On tool-use, paper-faithful SDPO peaks early (peak@50 = peak@150 because the cliff hits before step 60) and then collapses on the training set. CPSD-LM reaches a higher peak than SDPO and, as Figure 1a shows, its trajectory remains stable past step 60. End-of-training accuracy is no longer divorced from the peak; removing the cliff is what makes the 150-step peak a meaningful deployment metric rather than an artefact of stopping at the right moment.

Highest peak on every task (Qwen3-8B). At the fixed 150-step budget, CPSD-LM attains the best peak@150 in every column of the Qwen3-8B block of Table 1: tool-use 69.49 (+2.5 over GRPO, +2.0 over SDPO), biology 67.12 (+3.6 over the SDPO 63.50 ceiling that previously dominated this task), chemistry 81.73 (+11.9 over GRPO, +3.6 over SDPO), physics 72.50 (+0.9 over GRPO), material 80.32 (narrowly above SDPO 80.19), and LCB-v6 65.16 (+2.1 over GRPO 63.09, +3.9 over SDPO 61.29).

5.3 Training dynamics across tasks

Figure 3 extends the tool-use trajectory of Figure 1a to the four SciKnowEval domains. The late-training drop visible on tool-use recurs in milder forms on biology, physics, and material; CPSD-LM is the only method whose peak-end gap stays close to zero across all four.

6 Analysis and Ablations

We isolate the contribution of each CPSD lever.

Factorial 2×2: picker vs. loss form. Table 2 separates Lever 1 (picker) from Lever 3 (loss form) on tool-use. Switching from a random in-group winner to the SEM picker contributes +2.30 pp (JS column) and +2.09 pp (ListMLE column); swapping

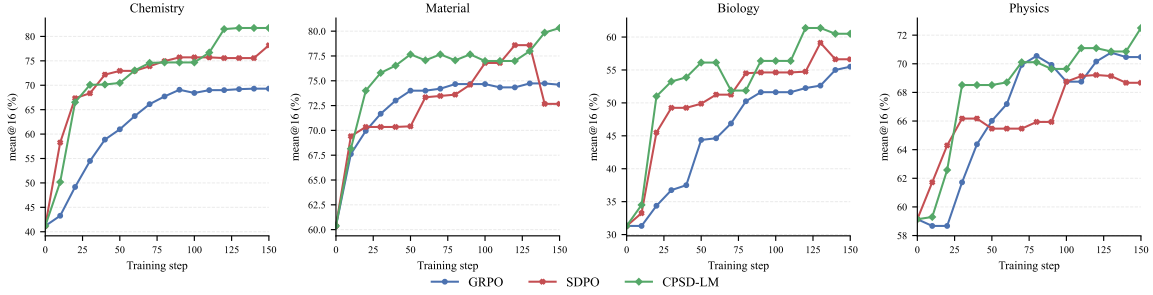


Figure 3: **Validation mean@16 over training on the four SciKnowEval domains** (Qwen3-8B, eval every 10 steps; 3-point median smoothed). CPD-LM attains the highest peak on every panel; on material it also avoids the late SDPO drop, and on physics it is the only method whose end value equals its peak.

Table 2: **Factorial 2×2 ablation on tool-use** (mean@16 peak@150, 150 steps, Qwen3-8B). Picker (Axis 1) and in-patch loss form (Axis 2) vary; all cells share the same adaptive-stop patch localisation, differing only in which winner the patch is anchored to (random vs SEM) and which loss runs inside the patch (JS vs ListMLE). Both axes contribute additively, and the combination (SEM picker, ListMLE patch) attains the best peak.

	JS patch	ListMLE patch (ours)
random winner	65.50	67.40
SEM picker (ours)	67.80	69.49

Table 3: **Q2: adaptive patch vs. global loss**. Peak@150 mean@16 with the CPD-LM stack on Qwen3-8B. *Global* reproduces SDPO’s whole-response distillation; *adaptive patch* applies ListMLE inside the SEM-anchored adaptive-stop window (Sec. 4.2).

Scope	Tool-use	Bio.	Chem.
global (entire response)	68.04	64.92	79.42
adaptive patch (ours)	69.49	67.12	81.73

the in-patch loss from JS to ListMLE contributes +1.90 pp (random row) and +1.69 pp (SEM row). Both axes are positively additive, and CPD-LM (SEM + ListMLE) attains the highest peak: removing either lever forfeits 1.7–3.99 pp.

Picker non-triviality. To confirm that the SEM picker is not a relabelling of the syntactic prefix-longest baseline, we compute, on $5 \times 20 = 100$ prompts $\times$ 8 base-model rollouts per task, the disagreement rate between the two pickers (Fig. 4). Disagreement is 44–65% across tasks; when the two pickers select different winners, the chosen winners differ by mean cosine distance 0.03–0.05 in the model’s embedding space, indicating SEM is reaching for a semantically distinct trajectory that prefix-matching does not surface.

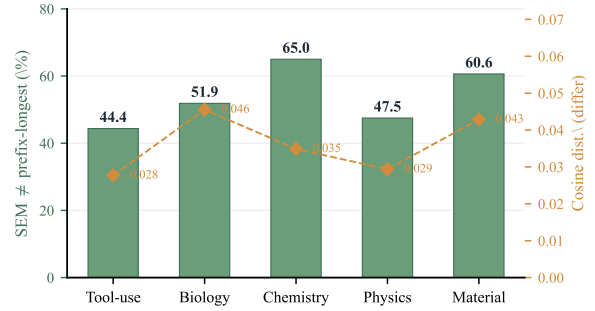


Figure 4: **SEM vs prefix-longest picker**. Bars: percent of in-group losers on which the two pickers select different winners (160 losers per task). The number under each task is the mean cosine distance between the two pickers’ choices in embedding space, computed only over the disagreement subset.

Patch vs. global. Table 3 contrasts the adaptive-stop patch (Sec. 4.2) with the global whole-response loss used by SDPO, holding all other components of CPD-LM fixed. Localising the distillation loss to the adaptive-stop window improves peak@150 by +1.5 pp on tool-use, +2.2 pp on biology, and +2.3 pp on chemistry. The global variant penalises tokens the loser had already gotten right and re-introduces the over-correction failure described in Section 3; the patch supplies gradient only at the decision point and stops once the student has locally re-aligned with the teacher.

First-divergence geometry. Figure 5 plots, per task, the empirical distribution of the first-divergence position t^* between pairs of base-model rollouts. Across five benchmarks the medians cluster between 8 and 11 tokens, and the mode is at $t^* < 8$ on every task. Two consequences: (i) the “decision point” is consistently near the start of the response, not deep in the chain of thought, so localising distillation at t^* captures the structurally informative region; (ii) the long tail ($\sim 10\%$ of pairs have $t^* > 32$ on chemistry, $\sim 20\%$ on tool-use) jus-

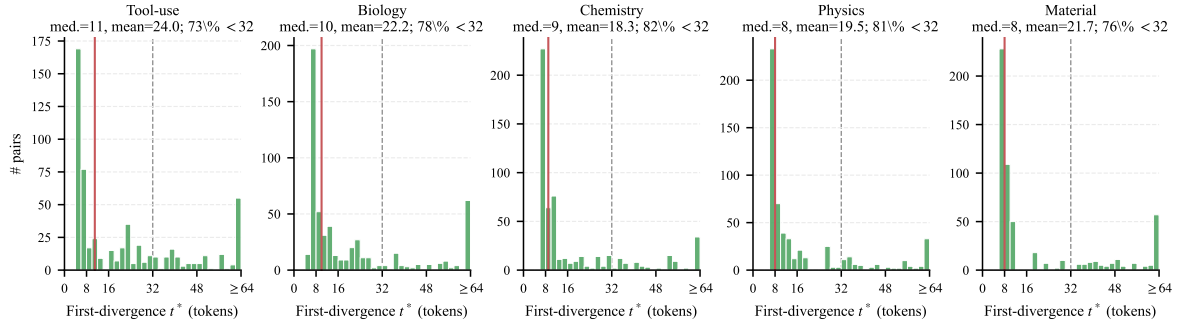


Figure 5: **First-divergence position t^* across five tasks** (base Qwen3-8B, 20 prompts $\times$ 8 in-group rollouts, $\sim$ 560 pairs per task). Bins are 2 tokens; the rightmost bin aggregates all $t^* \geq 64$. The distribution is heavily right-skewed: medians lie at 8–11 tokens and 73–82% of rollout pairs diverge within the first 32 tokens. First-divergence is therefore a well-defined early geometric anchor; localising distillation at t^* targets exactly the decision point while leaving the (typically much longer) downstream completion unconstrained.

tifies the adaptive cap rather than a hard short-fixed length, since occasional late divergences still benefit from a longer patch.

Geometry-probe diagnosis (Figure 7, appendix). An instrumented CPSD training run logs per-batch divergence-geometry metrics: SEM non-lexical winner %, mean first-divergence position, patch pairs per batch, first/random JS-mass ratio, ratio-clip firing fraction, and raw GRPO loss. Across tasks these vary by an order of magnitude; in particular, biology fires $5\times$ more patches at $6\times$ the per-batch GRPO loss magnitude, structurally requiring a lower c , which fixed nominal hyper-parameters do not deliver. We present this as a diagnostic device, not a per-task scheduler.

7 Related Work

Self-distillation in RLVR. SDPO (Hübotter et al., 2026) is our immediate baseline: in-group self-distillation with a global JS loss. Several recent works diagnose related failure modes: OPSD (Zhao et al., 2026d) argues that even slow EMA teachers degrade reasoning via epistemic suppression and recommends a frozen-at-init teacher; Kim et al. (2026b) show that EMA rates as low as 0.05 suppress “wait/hmm”-style epistemic verbalisation by up to 40% out-of-domain. Concurrent work on self-distillation RLVR includes RLRT (Kim et al., 2026a) (reverse teacher signal on correct rollouts), OGLS-SD (Yang et al., 2026) (outcome-contrast teacher), AOPD (Jia et al., 2026) (advantage-gated FKL), SDAR (Lu et al., 2026) (sigmoid-gated distillation), RLAD (Zhang et al., 2026) (TRRD blends student/teacher ratios), AMR-SD (Wei et al., 2026) (adaptive multi-rollout), Sample-Routing SDPO (Li et al., 2026a) (unifies GRPO and SDPO

at sample granularity), Entropy-Aware OPD (Jin et al., 2026), PCSD (Lv et al., 2026) (persistent consistency for self-distillation in agentic RL), and contrastive reinforced policy optimization with privileged self-distillation (Wu et al., 2026), and ROSD (Zhao et al., 2026e) (reflective on-policy self-distillation across domains), and Uni-OPD (Hou et al., 2026) (a dual-perspective recipe for on-policy distillation). CPSD is the *localised, rank-target* variant: it does not invent a new teacher signal but corrects the SDPO loss *where* and *how* it acts.

Listwise distillation and learning-to-rank.

ListMLE (Xia et al., 2008) and the Plackett–Luce model it instantiates predate LLMs by decades; in the LLM era, listwise targets have appeared in offline preference optimisation (LiPO, Liu et al., 2025a; LIRE, Zhu et al., 2024; PLD, Bassam et al., 2026; LPO, Qu et al., 2026). To our knowledge, CPSD is the first to use a rank-only listwise loss *inside an on-policy RLVR localised counterfactual patch*. The combination is what is new: not the loss, not the patch, but a patch-target that preserves multi-mode reasoning while still correcting the precise decision point.

8 Conclusion

Self-distillation in RLVR collapses for three closely related reasons: the canonical method picks a teacher essentially at random, applies a global distillation loss that over-corrects correctly-generated intermediate tokens, and uses a JS divergence target that is rigid enough to crash student diversity on multi-mode reasoning tasks. CPSD addresses each cause with a small, composable change: a SEM embedding-edit-distance picker, an adaptive-

stop first-divergence patch window, and a Plackett–Luce ListMLE rank loss inside the window. The combined method matches or exceeds GRPO and SDPO on all six reasoning benchmarks at the same compute budget on Qwen3-8B and on five of six tasks on Qwen3-4B; on chemistry, physics, material, and LCB-v6 it attains end-of-training accuracy that equals its peak, where SDPO drops 7–60 pp. We show that *where and how* self-distillation is applied inside RLVR matters at least as much as the choice of teacher, and that rank-only targets are a simple and underappreciated stability mechanism.

Limitations

Scale. Results cover Qwen3-8B and Qwen3-4B; smaller (0.5B) and larger (32B) backbones, as well as model families other than Qwen3, are untested.

Observational probe. The first-divergence geometry probe (Appendix F) is observational. We do not construct a controlled synthetic environment where patch geometry can be intervened while reward is held fixed; the case for adaptive-stop localisation rests on the patch-vs-global comparison (Table 3) rather than on a controlled causal study.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (92570101) and the Earth System BigData Platform of the School of Earth Sciences, Zhejiang University.

References

- Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pages 21246–21263.
- Ejafa Bassam, Dawei Zhu, and Kaigui Bian. 2026. Pld: A choice-theoretic list-wise knowledge distillation. *Advances in Neural Information Processing Systems*, 38:136090–136112.
- Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359.
- Kehua Feng, Xinyi Shen, Weijie Wang, Xiang Zhuang, Yuqi Tang, Qiang Zhang, and Keyan Ding. 2024. Sci-knoweval: Evaluating multi-level scientific knowledge of large language models. *arXiv preprint arXiv:2406.09098*.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Yu Guo, Dong Jin, Shenghao Ye, Shuangwu Chen, Jian Yang, and Xiaobin Tan. 2025b. Sqlforge: Synthesizing reliable and diverse data to enhance text-to-sql reasoning in llms. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8441–8452.
- Yu Guo, Shenghao Ye, Shuangwu Chen, Zijian Wen, Tao Zhang, Bai Qirui, Dong Jin, Yunpeng Hou, Huasen He, Xiaobin Tan, and 1 others. 2026. Rethinking table pruning in tableqa: From sequential revisions to gold trajectory-supervised parallel search. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12960–12976.
- Zhezheng Hao, Hong Wang, Haoyang Liu, Jian Luo, Jiarui Yu, Hande Dong, Qiang Lin, Can Wang, and Jiawei Chen. 2026a. Rethinking entropy interventions in rlvr: An entropy change perspective. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31105–31133.
- Zhezheng Hao, Hong Wang, Jian Luo, Jianqing Zhang, Yuyan Zhou, Qiang Lin, Can Wang, Hande Dong, and Jiawei Chen. 2026b. Recreate: Reasoning and creating domain agents driven by experience. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31018–31046.
- Zhezheng Hao, Tianfu Wang, Huanshuo Dong, Ziyang Liu, Hong Wang, Xiankun Lin, Qiang Lin, Can Wang, Hande Dong, and Jiawei Chen. 2026c. Evolve as a team: Collaborative self-evolution for llm-based multi-agent systems. *arXiv preprint arXiv:2605.29790*.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
- Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, and 1 others. 2026. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *arXiv preprint arXiv:2605.03677*.
- Junan Hu, Jian Liu, Jingxiang Lai, Jiarui Hu, Yiwei Sheng, Shuang Chen, Jian Li, Dazhao Du, and Song Guo. 2026. Gui agents with reinforcement learning: Toward digital inhabitants. *arXiv preprint arXiv:2604.27955*.
- Jonas Hübötter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Büning,

- Carlos Guestrin, and 1 others. 2026. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*.
- Naman Jain, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2025. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pages 58791–58831.
- Nan Jia, Haojin Yang, Xing Ma, Jiesong Lian, Shuailiang Zhang, Weipeng Zhang, Ke Zeng, Xunliang Cai, and Zequn Sun. 2026. Asymmetric on-policy distillation: Bridging exploitation and imitation at the token level. *arXiv preprint arXiv:2605.06387*.
- Woogyel Jin, Taywon Min, Yongjin Yang, Swanand Ravindra Kadhe, Yi Zhou, Dennis Wei, Nathalie Baracaldo, and Kimin Lee. 2026. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*.
- Jeonghye Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. 2026a. Rebellious student: Reversing teacher signals for reasoning exploration with self-distilled rlvr. *arXiv preprint arXiv:2605.10781*.
- Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. 2026b. Why does self-distillation (sometimes) degrade the reasoning capability of llms? *arXiv preprint arXiv:2603.24472*.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Gengsheng Li, Tianyu Yang, Junfeng Fang, Mingyang Song, Mao Zheng, Haiyun Guo, Dan Zhang, Jinqiao Wang, and Tat-Seng Chua. 2026a. Unifying group-relative and self-distillation policy optimization via sample routing. *arXiv preprint arXiv:2604.02288*.
- Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, and 1 others. 2026b. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*.
- Tianqi Liu, Zhen Qin, Junru Wu, Jiaming Shen, Misha Khalman, Rishabh Joshi, Yao Zhao, Mohammad Saleh, Simon Baumgartner, Jialu Liu, and 1 others. 2025a. Lipo: Listwise preference optimization through learning-to-rank. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2404–2420.
- Yule Liu, Heyi Zhang, Jinyi Zheng, Zhen Sun, Zifan Peng, Jiaheng Wei, Tianshuo Cong, Yilong Yang, and Xinlei He. 2025b. Auditing data membership in reinforcement learning with verifiable rewards. *arXiv preprint arXiv:2511.14045*.
- Yule Liu, Jinyi Zheng, Zhen Sun, Zifan Peng, Wenhan Dong, Zeyang Sha, Shiwen Cui, Weiqiang Wang, and Xinlei He. 2025c. Thought manipulation: External thought can be efficient for large reasoning models. *arXiv preprint arXiv:2504.13626*.
- Ziyan Liu, Zhezheng Hao, Yeqiu Chen, Hong Wang, Jingren Hou, Ruiyi Ding, Yongkang Yang, Wence Ji, Wei Xia, and Feng Liu. 2026. Meta-cognitive memory policy optimization for long-horizon llm agents. *arXiv preprint arXiv:2605.30159*.
- Yunbo Long, Yuhang Liu, and Liming Xu. 2026a. Emomas: Emotion-aware multi-agent system for high-stakes edge-deployable negotiation with bayesian orchestration. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31737–31773.
- Yunbo Long, Haolang Zhao, Lukas Beckenbauer, Liming Xu, and Alexandra Brintrup. 2026b. Emodistill: Offline emotion skill distillation for language model agents in adversarial negotiation. *arXiv preprint arXiv:2605.26785*.
- Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and 1 others. 2026. Self-distilled agentic reinforcement learning. *arXiv preprint arXiv:2605.15155*.
- Chunji Lv, Yangguang Wei, Junlin Liu, Yang Gao, Ming Liu, Xinming Wang, Jinyang Wu, Guoren Wang, and Changsheng Li. 2026. Pcsd: Persistent consistency for self-distillation in agentic reinforcement learning. *arXiv preprint arXiv:2608.01837*.
- Yun Qu, Qi Wang, Yixiu Mao, Heming Zou, Yuhang Jiang, Yingyue Li, Wutong Xu, Lizhou Cai, Weijie Liu, Clive Bai, and 1 others. 2026. Listwise policy optimization: Group-based rlvr as target-projection on the llm response simplex. *arXiv preprint arXiv:2605.06139*.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the*

- Twentieth European Conference on Computer Systems*, pages 1279–1297.
- Yuqiao Tan, Minzheng Wang, Shizhu He, Huanxuan Liao, Chengfeng Zhao, Qiunan Lu, Tian Liang, Jun Zhao, and Kang Liu. 2025. Bottom-up policy optimization: Your language model policy secretly contains internal policies. *arXiv preprint arXiv:2512.19673*.
- Yuqiao Tan, Minzheng Wang, Bo Liu, Zichen Liu, Tian Liang, Shizhu He, Jun Zhao, and Kang Liu. 2026. From $p(y|x)$ to $p(y)$: Investigating reinforcement learning in pre-train space. *arXiv preprint arXiv:2604.14142*.
- Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.
- Can Wang, Yuhao Wang, Yushe Cao, Canran Xiao, and Fei Shen. 2026a. Lap-forensics: Latent-pixel consistency guided multimodal reasoning for deepfake detection. *arXiv preprint arXiv:2607.25962*.
- Hong Wang, Zhezheng Hao, Jian Luo, Chenxing Wei, Yao Shu, Lei Liu, Qiang Lin, Hande Dong, and Jiawei Chen. 2026b. Scheduling your llm reinforcement learning with reasoning trees. In *International Conference on Learning Representations*, volume 2026, pages 154734–154753.
- Zhenlin Wei, Pu Jian, Yingzhuo Deng, Xiaohan Wang, Jiajun Chai, Zhexin Hu, Wei Lin, Shanbin Zhang, and Guojun Yin. 2026. Amr-sd: Asymmetric meta-reflective self-distillation for token-level credit assignment. *arXiv preprint arXiv:2605.18529*.
- Xingjian Wu, Junlin Liu, Xingchen Liu, Xuhang Zhu, Jianing Wang, Linsen Guo, Xiaoyu Li, Xuezhi Cao, and Xunliang Cai. 2026. Contrastive reinforced policy optimization via privileged self-distillation. *arXiv preprint arXiv:2607.28026*.
- Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: Theory and algorithm. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, pages 1192–1199.
- Can Xie, Ruotong Pan, Xiangyu Wu, Zhang Yunfei, Jiayi Fu, Tingting Gao, and Guorui Zhou. 2026a. Unlocking exploration in rlvr: Uncertainty-aware advantage shaping for deeper reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 19057–19076.
- Can Xie, Yuyi Zhou, Wen Yang, Siyao Song, Yingzhuo Deng, Shuo Ren, Jiajun Zhang, and 1 others. 2026b. Edge: Experience-distillation for guided exploration in agentic reinforcement learning. *arXiv preprint arXiv:2608.21946*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Yuxiao Yang, Xiaoyun Wang, and Weitong Zhang. 2026. Ogl-sd: On-policy self-distillation with outcome-guided logit steering for llm reasoning. *arXiv preprint arXiv:2605.12400*.
- Zhen Yao, Bo Lang, Zhihao Zheng, Haoyan Xu, Zhengtao Yao, Zhikang Dong, and Mooi Choo Chuah. 2026a. From semantic to reasoning: A survey of language-guided segmentation in the multimodal era. *TechRxiv*.
- Zhen Yao, Xin Li, Taotao Jing, Shuai Zhang, and Mooi Choo Chuah. 2026b. Disa: Saliency-aware foreground-background disentangled framework for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2601.20064*.
- Shenghao Ye, Yu Guo, Dong Jin, Yuxiang Wang, Yikai Shen, Yunpeng Hou, Shuangwu Chen, Xiaofeng Jiang, and 1 others. 2026a. When tableqa meets noise: A dual denoising framework for complex questions and large-scale tables. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24022–24045.
- Shenghao Ye, Yu Guo, Zhengheng Li, Shuangwu Chen, and Jian Yang. 2026b. Rubric-guided process reward for stepwise model routing. *arXiv preprint arXiv:2605.29310*.
- Shenghao Ye, Yuxiang Wang, Yu Guo, Dong Jin, Shuangwu Chen, and Jian Yang. 2026c. Rethinking stepwise model routing: A cost-efficient table reasoning perspective. *arXiv preprint arXiv:2605.29319*.
- Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gao-hong Liu, Lingjun Liu, and 1 others. 2026a. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244.
- Tianrun Yu, Yuxiao Yang, Zhaoyang Wang, Kaixiang Zhao, Porter Jenkins, Xuchao Zhang, Chetan Bansal, Huaxiu Yao, and Weitong Zhang. 2026b. Provable and practical in-context policy optimization for self-improvement. *arXiv preprint arXiv:2603.01335*.
- Tianrun Yu, Kaixiang Zhao, Chih-Chun Chen, Amanda Hughes, Taylor W Killian, Fenglong Ma, Weitong Zhang, and Porter Jenkins. 2026c. Lark: Learnability-grounded trajectory selection for efficient reasoning distillation. *arXiv preprint arXiv:2605.30651*.
- Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. 2026. Reinforcement-aware knowledge distillation for llm reasoning. *arXiv preprint arXiv:2602.22495*.

Haolang Zhao, Yunbo Long, Lukas Beckenbauer, and Alexandra Brintrup. 2026a. Veritrace: Evolving mental models for deep research agents. *arXiv preprint arXiv:2605.26081*.

Jinman Zhao, Erxue Min, Hui Wu, Ziheng Li, Zexu Sun, Hengyi Cai, Shuaiqiang Wang, Xu Chen, and Gerald Penn. 2026b. Beyond step pruning: Information theory based step-level optimization for self-refining large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 34941–34949.

Kaixiang Zhao, Tianrun Yu, Shawn Huang, Porter Jenkins, Yushun Dong, and Amanda Hughes. 2026c. Tiger: Traceable inference with graph-based evidence routing for mitigating hallucinations in multi-modal generation. *arXiv preprint arXiv:2606.00232*.

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

Ziqi Zhao, Xinyu Ma, Liu Yang, Yujie Feng, Daiting Shi, Jingzhou He, Xin Xin, Zhaochun Ren, and Xiaoming Wu. 2026e. Rosd: Reflective on-policy self-distillation for language model reasoning across domains. *arXiv preprint arXiv:2605.28014*.

Ziqi Zhao, Zhaochun Ren, Jiahong Zou, Liu Yang, Zhiwei Xu, Xuri Ge, Zhumin Chen, Xinyu Ma, Daiting Shi, Shuaiqiang Wang, and 1 others. 2026f. Reinforced efficient reasoning via semantically diverse exploration. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 47994–48007.

Dongzhuoran Zhou, Yuqicheng Zhu, Yule Liu, Zhen Yang, Rui Lu, Yuxiao Dong, Jie Tang, and Evgeny Kharlamov. 2026a. *Awm: Answerable working memory for long-document vqa agents*. *Preprint*, arXiv:2608.25618.

Dongzhuoran Zhou, Yuqicheng Zhu, Xiaxia Wang, Hongkuan Zhou, Yuan He, Jiaoyan Chen, Steffen Staab, and Evgeny Kharlamov. 2026b. What breaks knowledge graph based rag? benchmarking and empirical insights into reasoning under incomplete knowledge. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2522–2538.

Yixiao Zhou, Dongzhou Cheng, Zhiliang Wu, Yi Yang, Yu Cheng, and Hehe Fan. 2026c. One refiner to unlock them all: Inference-time reasoning elicitation via reinforcement query refinement. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 38957–38978.

Yixiao Zhou, Yang Li, Dongzhou Cheng, Hehe Fan, and Yu Cheng. 2026d. Look inward to explore outward: Learning temperature policy from llm internal states via hierarchical rl. *arXiv preprint arXiv:2602.13035*.

Mingye Zhu, Yi Liu, Lei Zhang, Junbo Guo, and Zhen-dong Mao. 2024. Lire: listwise reward enhancement for preference alignment. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3377–3394.

A Dataset Statistics

We use publicly available benchmarks with their standard training splits and evaluate on held-out validation or test splits. The training and held-out evaluation sizes, split names, and verifier types are listed below; the four SciKnowEval domains use the same multiple-choice verifier.

Task	Train	Held-out split	Verifier
Tool-use	4,046	validation (68)	JSON exact match
Biology	450	validation (50)	4-way MCQ
Chemistry	1,890	validation (210)	4-way MCQ
Physics	720	validation (80)	4-way MCQ
Material	841	validation (94)	4-way MCQ
LCB-v6	131	test (131)	Sandbox unit tests

Table 4: Training and held-out evaluation split statistics.

B Implementation Details

Codebase. Implementation on top of ver1 (Sheng et al., 2025) with FlashAttention-2 (Dao et al., 2022) and vLLM (Kwon et al., 2023). CPSD’s three levers are toggled by environment flags:

Lever (Section)	Environment flag(s)
SEM picker (Sec. 4.1)	SDPO_EDITDIST_WINNER=1
Adaptive-stop patch (Sec. 4.2)	SDPO_PATCH_ADDITIVE=1 SDPO_PATCH_ADAPTIVE_STOP=1 SDPO_PATCH_LEN=64 (cap)
ListMLE inside patch (Sec. 4.3)	SDPO_LOSS_MODE=rank_listmle

Default SDPO_LOSS_MODE=js recovers CPSD-JS.

Hardware and wall-clock. 4×H200, per-step training time (8-step warm average): tool-use 78.7 s, biology 91.0 s, chemistry 80.2 s, physics 84.4 s, material 72.8 s, LCB-v6 230 s (with parallel reward manager). Total to 150 steps: 3–6 hours per task.

Hyper-parameters. Identical across methods (top block) and CPSD-specific (bottom block):

C Patch-Coefficient Sweep

On biology, sweeping the patch coefficient $c \in \{0.04, 0.15, 0.25, 0.40, 0.50\}$ gives end accuracies

Parameter	Value
<i>Shared across GRPO / SDPO / CPSD</i>	
learning rate	3×10^{-6}
mini_batch	8
train_batch	32
rollout_n	8
distillation blend α	0.5
distillation_topk	100
is_clip	1.2
response_len	8192
<i>CPSD-specific (adaptive-stop patch)</i>	
SDPO_PATCH_ADAPTIVE_STOP	1
SDPO_PATCH_LEN (cap)	64
SDPO_PATCH_COEF (c)	0.25
teacher_update_rate (α_{EMA})	0.01
SDPO_PATCH_RATIO_CLIP_CAP (κ)	0.25

of $\{56, 60, 62.4, 60, 57\}$. The geometry-probe table predicts the optimum: biology fires patches $5\times$ more often than tool-use at $6\times$ the per-patch JS mass, so a fixed nominal c over-presses on biology. The ratio-clip safety net partially compensates but is by construction conservative.

D Teacher EMA Rate

On biology, $\alpha \in \{0, 0.01, 0.05\}$ yields end accuracies $\{55.4, 62.4, \text{collapse}\}$. Frozen teacher (the OPSD recommendation, Zhao et al., 2026d) drops 7 pp on our stack; fast EMA collapses. Slow EMA at 0.01 is the load-bearing middle. The cross-task mechanism is identical (cf. Kim et al., 2026b’s “epistemic suppression” under EMA teachers): on our stack, *some* teacher movement is required to keep up with the student.

E Supplementary Analyses

Peak-to-end-of-training stability (Fig. 6). For each (task, method) trained at the 150-step budget, we compute the gap between the peak and the last-step validation mean@16. CPSD keeps the gap below 12 pp on every task and below 7 pp on four of five; SDPO collapses 60.7 pp on tool-use and drifts 7.5 pp on material.

Divergence-geometry summary (Tab. 5). The combined per-task geometry numbers referenced throughout Sections 4 and 6.

F Geometry Probe

Figure 7 reports per-batch audit metrics over the first 60 training steps of an instrumented CPSD training run on each task. The five quantities are independently logged at every PPO microbatch and aggregated; cell values are row z -normalised for

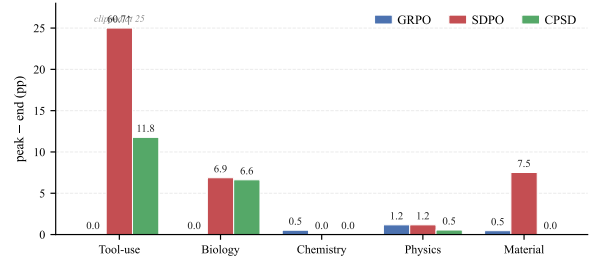


Figure 6: **Peak — end gap per (task, method).** Lower is more stable. Tool-use SDPO bar is clipped at 25 pp for legibility (true value 60.7); annotated values are raw.

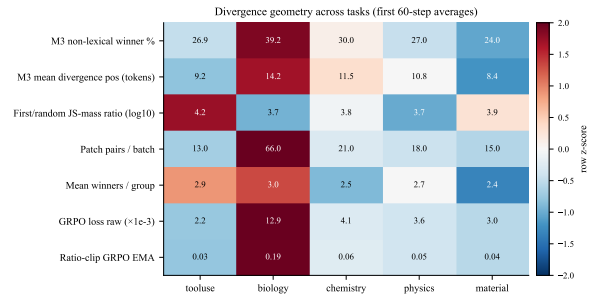


Figure 7: **Divergence-geometry probe across tasks** (first 60-step averages, row z -normalised; raw values printed in cells). Biology has roughly $5\times$ the patch-pair density and $6\times$ the GRPO loss of tool-use, yet ratio-clip rarely fires because the threshold scales with GRPO loss.

colour and the raw average is printed inside each cell.

G Additional Related Work

Beyond the self-distillation methods discussed in Section 7, recent work has explored various aspects of LLM reinforcement learning and post-training, including information-theoretic step-level optimization for self-refining LLMs (Zhao et al., 2026b) and rubric-guided process rewards for step-wise model routing (Ye et al., 2026b), in-context policy optimization for self-improvement (Yu et al., 2026b), and learnability-grounded trajectory selection for reasoning distillation (Yu et al., 2026c). Hao et al. (2026a) analyze entropy interventions in RLVR from an entropy-change perspective; uncertainty-aware advantage shaping has also been explored to promote deeper reasoning in RLVR (Xie et al., 2026a), alongside semantically diverse exploration for efficient reasoning (Zhao et al., 2026f). Related analyses audit data membership in reinforcement learning with verifiable rewards (Liu et al., 2025b) and study external thought manipulation for efficient reasoning (Liu et al., 2025c).

Table 5: **Divergence-geometry summary per task** (Qwen3-8B). *SEM $\neq$ pref.*: percent of in-group losers where the SEM picker chooses a different winner than the prefix-longest baseline (160 losers per task, base-model rollouts). *Median t^** : median first-divergence position in tokens. *Patches*: average loser/winner patch pairs per training batch. *JS ratio*: $\log_{10}$ of first-divergence JS-mass over a random-position window of equal width (training audit). *Ratio-clip*: average effective patch coefficient as a fraction of the GRPO loss budget. Biology stands out on patch density ($5\text{--}6\times$ tool-use) and ratio-clip firing.

Task	SEM $\neq$ pref. (%)	median t^*	patches/batch	JS ratio ($\log_{10}$)	ratio-clip (%)
Tool-use	44.4	11	13	4.19	3.20
Biology	51.9	10	66	3.67	18.80
Chemistry	65.0	9	21	3.84	6.20
Physics	47.5	8	18	3.66	5.40
Material	60.6	8	15	3.91	4.40

Other RL-based reasoning work explores inference-time query refinement (Zhou et al., 2026c) and hierarchical temperature policies driven by model internals (Zhou et al., 2026d). Wang et al. (2026b) propose scheduling strategies for LLM reinforcement learning using reasoning trees. Hao et al. (2026b) present experience-driven approaches for reasoning and creating domain agents; related work also studies reinforcement learning for GUI agents (Hu et al., 2026) and experience distillation for guided exploration in agentic RL (Xie et al., 2026b). Liu et al. (2026) introduce meta-cognitive memory policy optimization for long-horizon LLM agents; related work also develops answerable working memory for long-document VQA agents (Zhou et al., 2026a), while VeriTrace evolves mental models for deep-research agents (Zhao et al., 2026a). Hao et al. (2026c) explore collaborative self-evolution in multi-agent systems; related work also studies offline emotion-skill distillation for adversarial negotiation (Long et al., 2026b) and emotion-aware multi-agent negotiation (Long et al., 2026a). These works address complementary aspects of LLM post-training. Other analyses examine reinforcement learning in pre-training space (Tan et al., 2026) and bottom-up optimization of internal language-model policies (Tan et al., 2025). CPSD focuses specifically on the design of in-group self-distillation within single-turn RLVR.

Beyond verifier-based RLVR, related post-training and reasoning work also targets structured domains. Recent studies address noisy table question answering (Ye et al., 2026a), trajectory-supervised table pruning (Guo et al., 2026), cost-efficient stepwise routing for table reasoning (Ye et al., 2026c), and reliable data synthesis for text-to-SQL reasoning (Guo et al., 2025b), as well as knowledge-graph RAG under incomplete knowledge (Zhou et al., 2026b). These applica-

tions provide complementary examples of process supervision and step-level decision making outside the benchmark setting considered here; related evidence-routing work also studies hallucination mitigation in multimodal generation (Zhao et al., 2026c); other multimodal studies consider language-guided segmentation and saliency-aware foreground-background disentanglement (Yao et al., 2026a,b), as well as latent-pixel consistency for multimodal reasoning in deepfake detection (Wang et al., 2026a).

H Theoretical Properties

This appendix collects five formal results that underwrite the design choices in Section 4: the safety guarantee of the ratio-clip (Lemma 1), two invariances of the ListMLE objective (Propositions 2 and 3), the convergence behaviour of gradient flow under ListMLE (Theorem 4), and the non-interference of the localised patch with out-of-patch GRPO gradients (Lemma 5).

Notation. Throughout this appendix the total loss is $\mathcal{L}_{\text{total}} := \mathcal{L}_{\text{GRPO}} + c_{\text{eff}} \mathcal{L}_{\text{patch}}$ with c_{eff} defined by the ratio-clip rule of Lemma 1. As is standard in implementations of magnitude-based safety nets, c_{eff} enters the backward pass as a stop-gradient constant: it is recomputed at every step from the current $|\mathcal{L}_{\text{GRPO}}|$ and $|\mathcal{L}_{\text{patch}}|$ but is detached from the autograd graph, so $\partial c_{\text{eff}} / \partial \theta \equiv 0$. We adopt this convention in Lemma 5. The vector $\mathbf{s} = (s_1, \dots, s_K) \in \mathbb{R}^K$ collects student log-probabilities at a fixed token position over the teacher’s top- K tokens; S_K denotes the symmetric group on K symbols and $\sigma_T \in S_K$ is the teacher’s preference permutation (precise definition in Proposition 3).

H.1 Ratio-clip safety

Lemma 1 (Ratio-clip safety). *Let $c, \kappa > 0$ and $\mathcal{L}_{\text{patch}} \neq 0$. The effective patch coefficient*

$$c_{\text{eff}} = c \cdot \min\left\{1, \frac{\kappa |\mathcal{L}_{\text{GRPO}}|}{c |\mathcal{L}_{\text{patch}}|}\right\}$$

satisfies $|c_{\text{eff}} \mathcal{L}_{\text{patch}}| \leq \kappa |\mathcal{L}_{\text{GRPO}}|$.

Proof. We split on the value of the min (the two branches are equal at the boundary $c |\mathcal{L}_{\text{patch}}| = \kappa |\mathcal{L}_{\text{GRPO}}|$, where either case applies). If $\min = 1$ then $c |\mathcal{L}_{\text{patch}}| \leq \kappa |\mathcal{L}_{\text{GRPO}}|$ by definition of min, hence $|c_{\text{eff}} \mathcal{L}_{\text{patch}}| = c |\mathcal{L}_{\text{patch}}| \leq \kappa |\mathcal{L}_{\text{GRPO}}|$. If $\min = \kappa |\mathcal{L}_{\text{GRPO}}| / (c |\mathcal{L}_{\text{patch}}|)$ then $c_{\text{eff}} = \kappa |\mathcal{L}_{\text{GRPO}}| / |\mathcal{L}_{\text{patch}}|$ and so $|c_{\text{eff}} \mathcal{L}_{\text{patch}}| = \kappa |\mathcal{L}_{\text{GRPO}}|$. In either branch the inequality holds. $\square$

H.2 ListMLE invariances

For the remainder of this appendix we fix $K \geq 1$, write $\mathbf{s} \in \mathbb{R}^K$ and $\sigma \in S_K$, and study the loss

$$\mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \sigma) = \frac{1}{K} \sum_{i=1}^K \left[\log \sum_{j=i}^K e^{s_{\sigma(j)}} - s_{\sigma(i)} \right]. \quad (5)$$

Proposition 2 (Additive-shift invariance). *For every $c \in \mathbb{R}$ and $\sigma \in S_K$, $\mathcal{D}_{\text{ListMLE}}(\mathbf{s} + c \mathbf{1}; \sigma) = \mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \sigma)$.*

Proof. Fix rank position i . Under the substitution $\mathbf{s} \mapsto \mathbf{s} + c \mathbf{1}$,

$$\begin{aligned} & \log \sum_{j=i}^K e^{s_{\sigma(j)} + c} - (s_{\sigma(i)} + c) \\ &= \log \left(e^c \sum_{j=i}^K e^{s_{\sigma(j)}} \right) - s_{\sigma(i)} - c \\ &= \log \sum_{j=i}^K e^{s_{\sigma(j)}} - s_{\sigma(i)}, \end{aligned}$$

which is the original per-rank summand. Summing $i = 1, \dots, K$ completes the proof; the argument is unchanged for $K = 1$ (the sum has one term and is trivially preserved). $\square$

Proposition 3 (Teacher rank-only dependence). *Let $\mathbf{q}_t \in \mathbb{R}_{>0}^K$ be a teacher score vector with pairwise distinct entries, and let $\sigma_T := \arg \text{sort}(\mathbf{q}_t)$ be the unique permutation that sorts $\mathbf{q}_t$ in strictly descending order. For every strictly increasing $\phi : \mathbb{R} \rightarrow \mathbb{R}$ applied*

componentwise, $\arg \text{sort}(\phi(\mathbf{q}_t)) = \sigma_T$, and consequently $\mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \arg \text{sort}(\phi(\mathbf{q}_t))) = \mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \sigma_T)$.

Proof. The right-hand side of (5) references only $\mathbf{s}$ and σ , so the loss depends on $\mathbf{q}_t$ at most through the chosen permutation. For any indices a, b with $q_a \neq q_b$, strict monotonicity gives $\phi(q_a) > \phi(q_b) \iff q_a > q_b$; since the entries of $\mathbf{q}_t$ are pairwise distinct, every pair satisfies this hypothesis. The descending-sort permutation is therefore identical for $\mathbf{q}_t$ and $\phi(\mathbf{q}_t)$. $\square$

The role of σ_T in (5) is therefore purely combinatorial: under the distinctness hypothesis the loss is invariant to any teacher temperature, logit scale, or non-top- K residual mass. In practice the hypothesis is satisfied almost surely (under any continuous distribution on $\mathbf{q}_t$); ties are handled in implementations by a fixed tie-breaking rule applied uniformly across $\mathbf{q}_t$ and $\phi(\mathbf{q}_t)$, which preserves the equality of the resulting permutations.

H.3 Convergence of gradient flow

Theorem 4 (Gradient flow eliminates rank inversions). *Fix $\sigma_T \in S_K$ and consider the continuous gradient flow*

$$\dot{\mathbf{s}}(t) = -\nabla_{\mathbf{s}} \mathcal{D}_{\text{ListMLE}}(\mathbf{s}(t); \sigma_T),$$

with $\mathbf{s}(0) \in \mathbb{R}^K$. Call a pair (a, b) with $a < b$ a teacher-order inversion of $\mathbf{s}(t)$ if $s_{\sigma_T(a)}(t) < s_{\sigma_T(b)}(t)$. Then

1. *The number of teacher-order inversions of $\mathbf{s}(t)$ is non-increasing in t .*
2. *Every inversion present at $t = 0$ is resolved at some finite time $t^* < \infty$.*

Proof. Global existence. Each per-component gradient of the loss is a difference of softmax probabilities (computed below), so ∇L is smooth and componentwise bounded by 1 on $\mathbb{R}^K$. Local existence and uniqueness follow from Picard–Lindelöf; since the vector field is uniformly bounded, solutions admit no finite-time blow-up, hence extend to a unique global solution $\mathbf{s} \in C^1([0, \infty); \mathbb{R}^K)$.

Reduction to identity permutation. Let P_{σ_T} denote the coordinate permutation $(P_{\sigma_T} \mathbf{s})_i := s_{\sigma_T(i)}$. From equation (5), $\mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \sigma_T) = \mathcal{D}_{\text{ListMLE}}(P_{\sigma_T} \mathbf{s}; \text{id})$. The gradient flow on $\mathbf{s}$ is conjugate, under $\mathbf{r}(t) := P_{\sigma_T} \mathbf{s}(t)$, to the gradient flow on $\mathbf{r}$ for the loss $\mathcal{D}_{\text{ListMLE}}(\cdot; \text{id})$, because

P_{σ_T} is orthogonal and $\nabla(\mathcal{D} \circ P) = P^\top \nabla \mathcal{D} \circ P$. Moreover the teacher-order inversions of $\mathbf{s}$ correspond bijectively to the identity-order inversions of $\mathbf{r}$. We may therefore assume without loss of generality that $\sigma_T = \text{id}$; let $L(\mathbf{s}) := \mathcal{D}_{\text{ListMLE}}(\mathbf{s}; \text{id})$ and $Z_i(\mathbf{s}) := \sum_{j=i}^K e^{s_j}$, so that $L(\mathbf{s}) = \frac{1}{K} \sum_{i=1}^K [\log Z_i(\mathbf{s}) - s_i]$.

Step 1 — per-component gradient. For each $m \in \{1, \dots, K\}$,

$$\begin{aligned} \frac{\partial L}{\partial s_m}(\mathbf{s}) &= \frac{1}{K} \sum_{i=1}^K \left[\frac{e^{s_m} \mathbf{1}[m \geq i]}{Z_i(\mathbf{s})} - \mathbf{1}[m = i] \right] \\ &= \frac{1}{K} \left[\sum_{i=1}^m \frac{e^{s_m}}{Z_i(\mathbf{s})} - 1 \right]. \end{aligned}$$

Since $e^{s_m}/Z_i \leq 1$ for $m \geq i$, every component of ∇L lies in $[-1/K, (m-1)/K] \subset [-1, 1]$, confirming the boundedness used above.

Step 2 — pair-gap derivative. For $a < b$, let $\Delta_{ab}(t) := s_a(t) - s_b(t)$. Subtracting two copies of the formula in Step 1 and factoring e^{s_b} ,

$$\dot{\Delta}_{ab} = \frac{e^{s_b}}{K} \left[(1 - e^{\Delta_{ab}}) \sum_{i=1}^a \frac{1}{Z_i} + \sum_{i=a+1}^b \frac{1}{Z_i} \right]. \quad (6)$$

The factoring is valid for every $(s_a, s_b) \in \mathbb{R}^2$ since $e^{s_a} = e^{s_b} e^{\Delta_{ab}}$.

Step 3 — sign at $\Delta_{ab} \leq 0$. When $\Delta_{ab} \leq 0$ we have $1 - e^{\Delta_{ab}} \geq 0$, both bracketed terms in (6) are non-negative, and the second sum is over the non-empty range $\{a+1, \dots, b\}$ (since $a < b$), so whenever $\Delta_{ab}(t) \leq 0$,

$$\begin{aligned} \dot{\Delta}_{ab}(t) &\geq \frac{e^{s_b(t)}}{K} \sum_{i=a+1}^b \frac{1}{Z_i(\mathbf{s}(t))} \\ &> 0. \end{aligned} \quad (7)$$

Step 4 — the half-line $\Delta_{ab} \geq 0$ is forward-invariant. Suppose $\Delta_{ab}(0) \geq 0$. We claim $\Delta_{ab}(t) \geq 0$ for all $t \geq 0$. If not, by continuity of Δ_{ab} there exists a first crossing time $t^* > 0$ with $\Delta_{ab}(t^*) = 0$ and $\Delta_{ab}(t) < 0$ on some right-neighbourhood of t^* . But (7) applied at the crossing point gives $\dot{\Delta}_{ab}(t^*) > 0$, so by continuity of $\dot{\Delta}_{ab}$ there is $\varepsilon > 0$ with $\dot{\Delta}_{ab}(t) > 0$ on $[t^*, t^* + \varepsilon]$, contradicting the right-neighbourhood condition. Hence no inversion can be created from a non-inversion, and the inversion count is non-increasing. This proves (1) and also shows that once an inversion is resolved (i.e. Δ_{ab} reaches 0), it stays resolved.

Step 5 — finite-time resolution of an existing inversion. Suppose $\Delta_{ab}(0) < 0$. By (7), $\dot{\Delta}_{ab}(t) > 0$ as long as $\Delta_{ab}(t) < 0$, so on that interval Δ_{ab} is strictly monotone-increasing. Assume for contradiction that $\Delta_{ab}(t) < 0$ for every $t \geq 0$.

Each per-rank summand of L equals $\log(1 + \sum_{j>i} e^{s_j - s_i}) \geq 0$, so $L \geq 0$ pointwise. Gradient flow on a C^1 function satisfies $\frac{d}{dt} L(\mathbf{s}(t)) = -\|\nabla L(\mathbf{s}(t))\|^2 \leq 0$, hence $L(\mathbf{s}(t)) \leq L(\mathbf{s}(0)) =: L_0$ for all $t \geq 0$. The $i = a$ summand of $L(\mathbf{s}(t))$ is non-negative and at most $K \cdot L(\mathbf{s}(t)) \leq KL_0$, so

$$\log Z_a(\mathbf{s}(t)) - s_a(t) \leq KL_0,$$

or equivalently, with $C_1 := e^{KL_0}$,

$$Z_a(\mathbf{s}(t)) \leq C_1 e^{s_a(t)}.$$

For each $i \in \{a+1, \dots, b\}$ the partial sum $Z_i = \sum_{j=i}^K e^{s_j}$ has fewer terms than Z_a , so $Z_i(\mathbf{s}(t)) \leq Z_a(\mathbf{s}(t)) \leq C_1 e^{s_a(t)}$. Define the bracketed lower-bound integrand

$$B(t) := \frac{e^{s_b(t)}}{K} \sum_{i=a+1}^b \frac{1}{Z_i(\mathbf{s}(t))}.$$

For each $i \in \{a+1, \dots, b\}$, the inequality $Z_i \leq C_1 e^{s_a}$ gives

$$\begin{aligned} \frac{e^{s_b(t)}}{Z_i(\mathbf{s}(t))} &\geq \frac{e^{s_b(t)}}{C_1 e^{s_a(t)}} = \frac{e^{-\Delta_{ab}(t)}}{C_1} \\ &> \frac{1}{C_1}, \end{aligned}$$

the last inequality using $\Delta_{ab}(t) < 0 \Rightarrow e^{-\Delta_{ab}(t)} > 1$.

1. Summing over the $b - a \geq 1$ indices,

$$B(t) \geq \frac{b-a}{K C_1} \geq \frac{1}{K C_1} > 0, \quad (8)$$

holding uniformly for every $t \geq 0$ while $\Delta_{ab}(t) < 0$. By (7), $0 \leq B(t) \leq \dot{\Delta}_{ab}(t)$ on that interval. Integrating $\dot{\Delta}_{ab}$ from 0 to T and using $\Delta_{ab}(T) < 0$,

$$\begin{aligned} \int_0^T B(t) dt &\leq \int_0^T \dot{\Delta}_{ab}(t) dt \\ &= \Delta_{ab}(T) - \Delta_{ab}(0) \leq |\Delta_{ab}(0)|. \end{aligned}$$

Since $B \geq 0$ is continuous, the monotone convergence theorem as $T \rightarrow \infty$ gives $\int_0^\infty B(t) dt \leq |\Delta_{ab}(0)| < \infty$. But (8) forces $\int_0^\infty B(t) dt = \infty$, a contradiction. Hence $\Delta_{ab}(t^*) = 0$ at some finite t^* , and by Step 4 the inequality $\Delta_{ab}(t) \geq 0$ persists for all $t \geq t^*$, completing (2). $\square$

Remark (scope). Theorem 4 is an idealised result about gradient flow on the K -dimensional score vector $\mathbf{s}$ in isolation; it formalises why ListMLE is a suitable target for in-patch supervision (every teacher-order inversion is detected and eliminated in finite time, with no spurious inversions introduced). The full training dynamics under $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{GRPO}} + c_{\text{eff}} \mathcal{L}_{\text{patch}}$ involve additional gradient contributions through the GRPO term and the parameter-space mapping $\theta \mapsto \mathbf{p}_t$; the theorem isolates the rank-recovery property of the ListMLE component alone.

H.4 Patch localisation

Lemma 5 (Out-of-patch invariance). *Let $\mathcal{L}_{\text{patch}} = \sum_{\ell \in \mathcal{L}} \sum_{t \in T_{\text{patch}}^{(\ell)}} D_t$, where D_t depends on the student parameters θ only through $\mathbf{p}_t$. Assume in addition that $\mathcal{L}_{\text{GRPO}}$ depends on θ only through $\{\mathbf{p}_\tau\}_\tau$ (i.e. no direct differentiable dependence on θ outside the per-position distributions), and that c_{eff} is treated as a stop-gradient constant in backpropagation (see Notation, above). For any token position $t_0 \notin \bigcup_{\ell \in \mathcal{L}} T_{\text{patch}}^{(\ell)}$, the contribution to $\partial \mathcal{L}_{\text{total}} / \partial \theta$ flowing through $\mathbf{p}_t[t_0]$ equals the contribution from $\mathcal{L}_{\text{GRPO}}$ through $\mathbf{p}_t[t_0]$ alone.*

Proof. Because c_{eff} is detached from the autograd graph, its derivative with respect to θ vanishes in the backward pass; combined with the assumption that $\mathcal{L}_{\text{GRPO}}, \mathcal{L}_{\text{patch}}$ depend on θ only through $\{\mathbf{p}_\tau\}_\tau$, the chain rule gives

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \theta} = \sum_{\tau} \left[\frac{\partial \mathcal{L}_{\text{GRPO}}}{\partial \mathbf{p}_\tau} + c_{\text{eff}} \frac{\partial \mathcal{L}_{\text{patch}}}{\partial \mathbf{p}_\tau} \right] \cdot \frac{\partial \mathbf{p}_\tau}{\partial \theta}.$$

Since $\mathcal{L}_{\text{patch}}$ depends on $\mathbf{p}_\tau$ only through positions in $\bigcup_{\ell} T_{\text{patch}}^{(\ell)}$, $\partial \mathcal{L}_{\text{patch}} / \partial \mathbf{p}_{t_0} = 0$ whenever $t_0 \notin \bigcup_{\ell} T_{\text{patch}}^{(\ell)}$. The $\tau = t_0$ summand therefore reduces to $\frac{\partial \mathcal{L}_{\text{GRPO}}}{\partial \mathbf{p}_{t_0}} \cdot \frac{\partial \mathbf{p}_{t_0}}{\partial \theta}$, as claimed. $\square$

This formalises the statement in Section 4.2 that tokens outside the patch window receive no distillation gradient: at positions where the loser had already agreed with the chosen winner, the only training signal is GRPO’s own advantage-weighted update.

I Use of Large Language Models

Large language models were used only for language polishing and grammar checking. All manuscript content and experimental procedures and results were manually reviewed and verified

by the authors, who take full responsibility for the paper.