LIGHTNING OPD 2.0: MITIGATING STYLE BIAS IN CROSS-TEACHER ON-POLICY DISTILLATION FOR LARGE REASONING MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) provides dense token-level supervision from a teacher, but its effectiveness can depend on *teacher consistency*, meaning that the model providing OPD supervision should also have generated the demonstrations used to train the supervised fine-tuning (SFT) reference. However, this condition is frequently violated in practice when SFT data have mixed or unknown provenance or when different models are preferred for SFT data generation and subsequent distillation. In such cross-teacher settings, even a stronger OPD teacher can yield little improvement over the SFT reference. We find that raw teacher–reference disagreement contains potentially useful context-specific teacher evidence as well as a recurring component associated with differences in wording, formatting, and reasoning cadence. We introduce **Lightning OPD 2.0** with cross-fitted style residualization, which uses rollout-level cross-fitting to estimate this recurring component as an operational proxy for style-token bias and subtracts it before constructing the token-level OPD update. Across mathematical reasoning and code generation benchmarks, Lightning OPD 2.0 consistently outperforms Lightning OPD with three SFT references spanning Qwen and Olmo model families. Starting from Klear-Reasoner-8B-SFT, Lightning OPD 2.0 reaches 82.4% on AIME 2024 and 63.0% on LiveCodeBench v5. Together, these results establish Lightning OPD 2.0 as a practical approach to cross-teacher OPD, relaxing teacher consistency as a prerequisite and allowing the SFT data generator and distillation teacher to be selected independently. Code will be released upon publication.

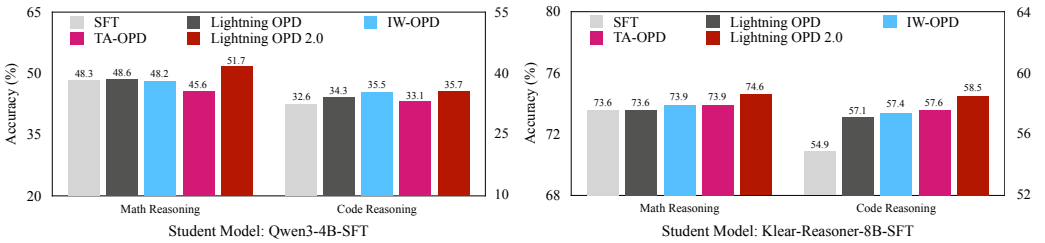


Figure 1: Average Pass@1 (%) on mathematical reasoning and code generation in the Qwen3-4B-SFT and Klear-Reasoner-8B-SFT settings with Qwen3-30B-A3B-Thinking-2507 as the OPD teacher. Lightning OPD 2.0 achieves the strongest average performance across both task families and both SFT references, consistently improving over Lightning OPD.

1 INTRODUCTION

On-policy distillation (OPD) has become an effective post-training alternative to reinforcement learning for large language models (LLMs) (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025; Yang et al., 2026a;c). Rather than learning from a sparse sequence-level reward, OPD uses a high-capability teacher to score student-generated trajectories and converts teacher–student disagreement into dense token-level supervision. This feedback provides guidance throughout the reasoning process, but standard OPD requires a live teacher throughout training.

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.

Lightning OPD (Wu et al., 2026) removes this systems bottleneck by caching rollouts from a supervised fine-tuning (SFT) reference policy and their teacher log-probabilities before optimization. A central finding is that effective distillation depends on *teacher consistency*: the OPD teacher should also have generated the SFT demonstrations. This condition aligns token-level supervision with the behavior already learned by the reference policy, allowing fully offline Lightning OPD to closely track standard online OPD.

Teacher consistency is difficult to satisfy when SFT data have mixed or undocumented provenance, leaving no single demonstration teacher to reuse. Even when the generator is known, testing a new OPD teacher should not require regenerating demonstrations and repeating SFT. More fundamentally, the best model for generating SFT demonstrations need not be the best teacher for subsequent distillation. Practitioners may use a stronger generator to obtain a high-quality SFT reference while selecting a different teacher for later optimization. Enforcing consistency then requires rebuilding the SFT pipeline or compromising one stage to accommodate the other. We study *cross-teacher OPD*, which preserves an existing SFT reference while allowing the OPD teacher to be selected independently. Naively switching teachers, however, can substantially degrade performance even when the replacement teacher is stronger.

To understand this degradation, we distinguish the roles of token-level disagreement. On reasoning-related tokens, a low teacher probability may identify an incorrect intermediate conclusion or operation and provide useful corrective supervision. On style tokens, the same low probability may instead reflect preferences for different wording, transitions, formatting, or reasoning cadence in an otherwise valid response. The raw OPD update treats both alike. Because style choices recur throughout a response, their accumulated penalties can systematically interfere with problem-specific reasoning supervision. This distinction suggests an observable proxy: style-related disagreement tends to recur across unrelated rollouts through similar lexical choices, response positions, and levels of reference-policy surprisal, whereas reasoning-related disagreement depends more strongly on the current problem and reasoning state. We therefore use the component predictable across cached rollouts as an operational proxy for style-token bias and the residual as a candidate carrier of context-specific teacher evidence.

Based on this observation, we introduce **Lightning OPD 2.0** with cross-fitted style residualization. We compute teacher–reference log-probability differences and estimate their recurring component using two lookup tables: one indexed by token identity, the other by normalized response position and reference-policy surprisal. Rollout-level cross-fitting constructs each token's estimate from other folds, excluding responses to the same prompt. This estimates recurring disagreement without fitting the correction to the token's own response. We average the two lookup predictions and subtract this estimate before applying the Lightning OPD update. The correction is computed once from the cache, preserving the fully offline training pipeline.

With Qwen3-30B-A3B-Thinking-2507 as the OPD teacher, we evaluate Qwen3-4B-SFT, Klear-Reasoner-8B-SFT, and Olmo-3-7B-Think-SFT on mathematical reasoning and code generation. Lightning OPD 2.0 improves over Lightning OPD by 3.1, 1.0, and 2.4 points on average math performance and 1.4, 1.4, and 2.8 points on average code performance, respectively. Olmo extends these gains across model families and tokenizers. With Klear-Reasoner-8B-SFT, Lightning OPD 2.0 reaches 82.4% on AIME 2024 and 63.0% on LiveCodeBench v5.

Our contributions are threefold.

- We formulate the *cross-teacher* OPD setting, in which the SFT data generator and the OPD teacher are selected independently, and show that distillation effects can substantially degrade under this setting.
- We identify a recurring component of teacher–reference disagreement that is predictable across rollouts and associated with style-token bias, explaining why raw cross-teacher supervision can be misleading.
- We introduce Lightning OPD 2.0, which estimates this recurring component with cross-fitted token and context lookup tables and subtracts it before applying the Lightning OPD update. Across three cross-teacher settings, Lightning OPD 2.0 consistently improves Lightning OPD on mathematical reasoning and code generation benchmarks and achieves state-of-the-art performance.

2 Related Work

LLM Post-Training. Post-training is central to capable LLMs, commonly combining supervised fine-tuning on high-quality instruction and reasoning demonstrations (Ouyang et al., 2022; Guha et al., 2026; Yang et al., 2024; Guo et al., 2025) with preference- and reward-based optimization (Schulman et al., 2017; Rafailov et al., 2023; Li et al., 2024; Dong et al., 2023). For reasoning, recent methods develop outcome-based policy optimization (Shao et al., 2024; Liu et al., 2025b; Yu et al., 2025; Zheng et al., 2025; MiniMax et al., 2025) and improve process-level credit assignment or value learning (Cui et al., 2026; Kazemnejad et al., 2025; Yuan et al., 2025; Yue et al., 2025b). These advances support increasingly capable mathematical and general reasoning systems (Guo et al., 2025; DeepSeek-AI et al., 2025; Kimi Team et al., 2025; Hu et al., 2025; Yang et al., 2024; 2025; Wang et al., 2026a; Yang et al., 2026c; NVIDIA et al., 2025; LLM-Core Xiaomi et al., 2026; Kimi Team et al., 2026; GLM-5 Team et al., 2026). Complementary analyses study what online optimization adds beyond SFT and why it can better preserve existing capabilities (Yue et al., 2025a; Shenfeld et al., 2026b), while recent methods co-design or interleave SFT and RL (Yan et al., 2025; Ma et al., 2026; Zhang et al., 2026a; Chen et al., 2026; Liu et al., 2025a; Huang et al., 2026). Our work instead studies the coupling between the model that generates the SFT demonstrations and the teacher that provides subsequent OPD supervision, seeking to remove teacher consistency as a practical prerequisite for effective OPD.

On-Policy Distillation. Knowledge distillation transfers capabilities from a teacher to a student by matching their output distributions (Hinton et al., 2015). Sequence-level and language-model distillation extend this idea to autoregressive generation (Kim & Rush, 2016; Gu et al., 2024; Rang et al., 2025; Xu et al., 2025), whereas on-policy distillation evaluates the teacher on student-generated trajectories and provides dense feedback on states visited by the student (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025; Song & Zheng, 2026). Recent work extends OPD through reward extrapolation and flexible reference policies (Yang et al., 2026a), entropy-aware divergence and local-support matching (Jin et al., 2026; Fu et al., 2026; Ke et al., 2026), reinforcement-aware or verifier-guided objectives (Zhang et al., 2026b; Xu et al., 2026), controllable reasoning (Liang et al., 2026), and black-box, privileged, or self-distillation settings (Ye et al., 2026; Zhao et al., 2026; Shenfeld et al., 2026a; Hübotter et al., 2026; Penaloza et al., 2026; Tan & Hong, 2026; Yang et al., 2026b). OPD has also become a component of large-scale reasoning post-training pipelines (Yang et al., 2025; LLM-Core Xiaomi et al., 2026; Yang et al., 2026c). Complementary studies show that stronger teachers do not always yield better students and trace failures to thinking-pattern incompatibility, unreliable token-level guidance, and teacher-dependent gradient quality (Li et al., 2026; Fu et al., 2026; Armandpour et al., 2026; Xu et al., 2026; Yang et al., 2026b). Offline OPD variants avoid maintaining a live teacher during optimization by precomputing rollouts and their corresponding teacher log-probabilities before training (Rang et al., 2025; Wu et al., 2026). Building upon Lightning OPD, we study how to maintain effective distillation in cross-teacher settings, where the SFT data generator and the OPD teacher differ. Lightning OPD 2.0 uses rollout-level cross-fitting to estimate the recurring component of teacher–reference disagreement as an operational proxy for style-token bias and subtracts it before constructing the token-level OPD update.

3 Methodology

3.1 Cross-Teacher OPD

Let $\{q_i\}_{i=1}^{N}$ be the prompts stored in the replay. When the supervised fine-tuning (SFT) demonstrations have a single identifiable generator, let π_G denote this SFT data generator. Let π_R be the resulting SFT reference policy, π_T the model selected to provide OPD supervision, and π_θ a trainable policy initialized from π_R. Teacher consistency requires $\pi_T = \pi_G$ (Wu et al., 2026). We study *cross-teacher OPD*, where π_T is selected independently of π_G or no single π_G can be identified. Our method addresses this general setting through the frozen replay of Lightning OPD.

Following Lightning OPD, we sample and freeze one response $x_i = (y_{i1}, \ldots, y_{iL_i}) \sim \pi_R(\cdot \mid q_i)$ for every prompt. Let $h_{it} = (q_i, y_{i,<t})$ be the prefix at token t. With shared tokenization, the OPD teacher and the SFT reference score the same realized token, producing the cached chosen-token

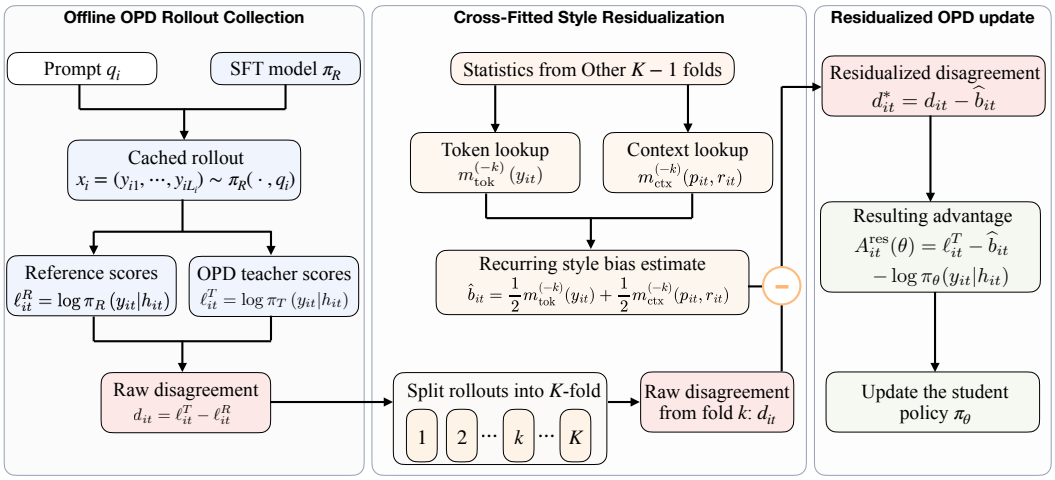


Figure 2: Overview of Lightning OPD 2.0. The SFT reference generates the frozen rollouts, which are scored by both the reference and the independently selected OPD teacher to obtain the raw teacher–reference disagreement. For each held-out fold, response-balanced token and context lookup tables fitted on the other folds estimate the recurring disagreement as an operational proxy for style-token bias. Subtracting this estimate yields a residualized signal for the reference-anchored Lightning OPD update.

log-probabilities

$$\ell_{it}^T = \log \pi_T(y_{it} \mid h_{it}), \qquad \ell_{it}^R = \log \pi_R(y_{it} \mid h_{it}). \tag{1}$$

In the Olmo–Qwen setting, cross-teacher distillation is also *cross-tokenizer*: the student and teacher use different tokenizers. We map each student response token to a byte-preserving teacher-token path and sum the conditional teacher log-probabilities along that forced path to obtain ℓ_{it}^T. The reference score ℓ_{it}^R remains the native student-token log-probability. Both scores are aligned to student response tokens, so residualization proceeds unchanged.

Using this cache, unmodified Lightning OPD retains the original per-token advantage

$$A_{it}^{\text{base}}(\theta) = \ell_{it}^T - \log \pi_\theta(y_{it} \mid h_{it}). \tag{2}$$

Following standard OPD practice (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025; Wu et al., 2026), we apply stop-gradient to the advantage when computing updates with the Lightning OPD policy surrogate. Although its value depends on the current student policy, the advantage enters the surrogate as a fixed weight for each parameter update.

To isolate the part of the signal affected by the teacher choice, we define the teacher–reference disagreement

$$d_{it} = \ell_{it}^T - \ell_{it}^R. \tag{3}$$

At initialization, $\pi_\theta = \pi_R$, so $A_{it}^{\text{base}} = d_{it}$. In cross-teacher OPD, d_{it} can contain both context-specific teacher evidence and recurring teacher–reference differences induced by the mismatch with the SFT data generator. The unmodified update cannot distinguish the two. The next subsection develops an operational proxy for this recurring component.

3.2 Predictability across Rollouts as a Style Proxy

The raw disagreement d_{it} can contain both context-specific teacher evidence and systematic differences in how the selected teacher and the SFT reference express a response. We make the operational assumption that disagreement associated with recurring style differences is more predictable across cached rollouts from simple lexical and coarse contextual coordinates, whereas potentially useful, reasoning-related teacher evidence is more dependent on the current context. We therefore write

$$d_{it} = b(z_{it}) + v_{it}, \tag{4}$$

where z_{it} collects the token and context coordinates defined below, $b(z_{it})$ is the component that recurs at those coordinates across rollouts, and v_{it} is the remaining variation. This is an operational decomposition rather than a semantic labeling of individual tokens. We use $b(z_{it})$ as a proxy for recurring style-token bias and approximate it by equally combining token- and context-based lookup predictions. We estimate these predictions by rollout-level cross-fitting and replace d_{it} with its estimated residual in the training objective.

Observable coordinates for recurring disagreement. Estimating a component that recurs across rollouts requires coordinates that are shared often enough to support reliable averaging but expressive enough to distinguish different uses of a token. Token identity captures lexical preference but treats every occurrence alike. At the other extreme, the full prefix identifies the local context precisely but rarely repeats across rollouts and may absorb problem-specific reasoning. We therefore complement token identity with a coarse context coordinate based on normalized response position and reference-policy surprisal. Normalized response position identifies the stage of the response. Reference-policy surprisal provides a low-dimensional summary of how typical the realized token is under the SFT reference. Together, these coordinates describe broad usage conditions without conditioning on the full prompt or prefix.

Formally, let the reference-policy surprisal be $\xi_{it} = -\ell_{it}^{R}$, the standard measure of how atypical a realized token is under a language model (Wilcox et al., 2023). A small value means that the token is natural under the SFT reference, while a large value means that the reference itself finds the token unexpected. We divide the response into B_{pos} normalized-position bins and the surprisal range into B_{ref} bins up to $\xi_{\max}$. Coarse surprisal bins distinguish disagreement on reference-typical and reference-atypical tokens while preserving sufficient support across rollouts for stable estimation. Their zero-indexed identifiers are

$$p_{it} = \left\lfloor B_{\mathrm{pos}} \frac{t-1}{L_i} \right\rfloor, \qquad r_{it} = \min\left\{ B_{\mathrm{ref}} - 1, \left\lfloor B_{\mathrm{ref}} \frac{\xi_{it}}{\xi_{\max}} \right\rfloor \right\}. \tag{5}$$

We set $z_{it}^{\mathrm{tok}} = y_{it}$, $z_{it}^{\mathrm{ctx}} = (p_{it}, r_{it})$, and $z_{it} = (z_{it}^{\mathrm{tok}}, z_{it}^{\mathrm{ctx}})$. Binning pools comparable occurrences across responses of different lengths. We construct the token and context lookup tables separately rather than using their full cross product, which avoids fragmenting the cache into sparse groups.

3.3 Cross-Fitted Estimation of Recurring Disagreement

Cross-fitting is a standard sample-splitting technique for constructing held-out predictions and preventing self-fitting (Chernozhukov et al., 2018; Guo et al., 2021). We partition the cached rollouts into K deterministic folds, keeping all rollouts originating from the same prompt in one fold. For each held-out fold k, all lookup statistics are estimated from the other $K-1$ folds. Neither a held-out response nor another response to the same prompt contributes to its lookup tables.

To prevent longer responses from dominating the estimates, we assign each response unit total weight, distributed uniformly over its active tokens, when fitting the lookup tables. For all methods, the training loss uses the same reduction by averaging active-token contributions within each response and then averaging across responses.

Using the non-held-out folds, we build one lookup table for the mean disagreement d_{it} of each token identity and another for each context coordinate. Rare groups are smoothed toward the global mean of the non-held-out folds, while unseen groups use that global mean directly. This prevents a token or context with few observations from receiving an unstable correction. Let $m_{\mathrm{tok}}^{(-k)}$ and $m_{\mathrm{ctx}}^{(-k)}$ denote the resulting smoothed lookup tables. For token (i,t) in held-out fold k, we use the equal-weight prediction

$$\widehat{b}_{it} = \frac{1}{2} m_{\mathrm{tok}}^{(-k)}(z_{it}^{\mathrm{tok}}) + \frac{1}{2} m_{\mathrm{ctx}}^{(-k)}(z_{it}^{\mathrm{ctx}}). \tag{6}$$

The two lookup predictions receive equal coefficients, and averaging keeps their combination on the scale of a single disagreement estimate. Equation (6) is our concrete approximation to $b(z_{it})$ in Equation (4). Because both lookup tables exclude the current prompt, $\widehat{b}_{it}$ captures recurring disagreement that is predictable from other cached rollouts rather than variation fitted specifically to the current response.

Algorithm 1 Lightning OPD 2.0 with Cross-Fitted Style Residualization

Require: SFT reference π_R, frozen replay $\mathcal{D}$ with scores ℓ^R and ℓ^T, and number of folds K
Compute d_{it} and the token and context coordinates defined in Section 3.2
Partition rollouts into K folds, grouping those from the same prompt
for each held-out fold k **do**
    Fit response-balanced, smoothed tables $m_{\text{tok}}^{(-k)}$ and $m_{\text{ctx}}^{(-k)}$ on the other folds
    Predict $\widehat{b}_{it}$ on fold k using Equation (6)
    Cache corrected teacher scores $\widetilde{\ell}_{it}^T = \ell_{it}^T - \widehat{b}_{it}$ for fold k
end for
Initialize $\pi_\theta \leftarrow \pi_R$
for each training step **do**
    Load a minibatch $\mathcal{B}$ from the corrected replay
    Compute A_{it}^{res} on $\mathcal{B}$ using Equation (9)
    Update θ with the Lightning OPD policy surrogate
end for
return π_θ

3.4 Residualized Lightning OPD

The base advantage separates into the cross-teacher disagreement and a reference-anchoring term

$$A_{it}^{\text{base}}(\theta) = d_{it} + \ell_{it}^R - \log \pi_\theta(y_{it} \mid h_{it}). \tag{7}$$

Cached teacher and reference scores fix d_{it}; the reference-anchoring term tracks the student's token-level deviation from its SFT initialization. We therefore remove the estimated recurring component $\widehat{b}_{it}$ from d_{it} while preserving the reference-anchoring term:

$$d_{it}^* = d_{it} - \widehat{b}_{it}. \tag{8}$$

The resulting advantage is

$$\begin{aligned} A_{it}^{\text{res}}(\theta) &= d_{it}^* + \ell_{it}^R - \log \pi_\theta(y_{it} \mid h_{it}) \\ &= \ell_{it}^T - \widehat{b}_{it} - \log \pi_\theta(y_{it} \mid h_{it}). \end{aligned} \tag{9}$$

At initialization, $\pi_\theta = \pi_R$ and $A_{it}^{\text{res}} = d_{it}^*$. Training uses A_{it}^{res} in place of A_{it}^{base} in the same policy surrogate. Equivalently, the fixed effective chosen-token teacher score $\widetilde{\ell}_{it}^T = \ell_{it}^T - \widehat{b}_{it}$ replaces ℓ_{it}^T. This score and $\widehat{b}_{it}$ are precomputed and fixed throughout optimization. Residualization thus preprocesses the frozen replay without further teacher scoring during student optimization. Algorithm 1 summarizes the procedure. Appendices B and C give the bounded implementation of the estimator and the policy surrogate.

4 Experiments

We describe the three cross-teacher settings and their protocols, report mathematical reasoning and code generation results, analyze style-token bias against teacher-consistent signals, and ablate token-identity lookup, context lookup, and prompt-level cross-fitting.

4.1 Experimental Setup

Cross-teacher settings. We study three complementary settings. The Qwen3-4B setting starts from the Qwen3-4B SFT reference used by Lightning OPD, whose SFT demonstrations were generated by Qwen3-8B, and selects Qwen3-30B-A3B-Thinking-2507 as the OPD teacher. Its known provenance makes the source of cross-teacher style mismatch directly identifiable. The second setting uses Klear-Reasoner-8B-SFT (Kwai-Klear, 2025), which is trained from Qwen3-8B-Base using long-chain-of-thought SFT data distilled from DeepSeek-R1-0528 (DeepSeek-AI, 2025). It tests whether the same correction applies to a stronger SFT reference trained from a different initialization and data generator. The third setting uses Olmo-3-7B-Think-SFT (Olmo Team et al., 2026) and is both

Table 1: Pass@1 (%) on math and code. † marks external results under the source papers' settings. ◊ marks frozen-replay adaptations; – indicates unavailable entries. **Bold** marks the best among our runs in each setting; blue highlights Lightning OPD 2.0.

Method	Math Reasoning				Code Generation		
	AIME 2024	AIME 2025	HMMT Feb. 2025	Avg.	LCB v5	LCB v6	Avg.
Student: Qwen3-4B-SFT Teacher: Qwen3-30B-A3B-Thinking-2507							
SFT reference	57.5	52.4	34.9	48.3	33.8	31.5	32.6
GKD† (Agarwal et al., 2024)	58.1	50.4	29.8	46.1	–	–	–
ExOPD† (Yang et al., 2026a)	61.0	56.0	34.4	50.5	–	29.0	–
AOPD† (Jia et al., 2026)	61.0	54.4	32.1	49.2	–	–	–
Lightning OPD	59.6	51.6	34.6	48.6	35.5	33.2	34.3
IW-OPD◊ (Xie et al., 2026)	60.8	49.9	33.8	48.2	**37.2**	33.8	35.5
TA-OPD◊ (Wang et al., 2026b)	56.0	50.0	30.6	45.6	35.7	30.4	33.1
Lightning OPD 2.0	**63.5**	**55.0**	**36.7**	**51.7**	36.9	**34.4**	**35.7**
Student: Klear-Reasoner-8B-SFT Teacher: Qwen3-30B-A3B-Thinking-2507							
Qwen3-8B	76.0	67.3	44.7	62.7	57.5	48.4	53.1
SFT reference	81.3	77.5	62.2	73.6	58.5	51.3	54.9
Lightning OPD	80.6	77.2	62.9	73.6	61.6	52.6	57.1
IW-OPD◊	81.9	77.6	62.3	73.9	61.0	53.8	57.4
TA-OPD◊	**82.7**	76.9	62.0	73.9	62.1	53.1	57.6
Lightning OPD 2.0	82.4	**77.7**	**63.8**	**74.6**	**63.0**	**53.9**	**58.5**
Student: Olmo-3-7B-Think-SFT Teacher: Qwen3-30B-A3B-Thinking-2507							
SFT reference	70.6	58.4	37.4	55.5	46.5	40.1	43.3
Lightning OPD	71.4	57.7	37.6	55.6	44.3	39.3	41.8
Lightning OPD 2.0	**72.4**	**61.2**	**40.3**	**58.0**	**48.2**	**41.0**	**44.6**

cross-teacher and *cross-tokenizer*: the Olmo student and Qwen teacher use different tokenizers. We retain Olmo's native tokenizer and chat template, aligning teacher scores to student tokens through the byte-preserving paths in Section 3.1. All three settings use Qwen3-30B-A3B-Thinking-2507 (Yang et al., 2025) as the selected cross-teacher OPD teacher.

Training Settings. We use DAPO-Math-17k (Yu et al., 2025) for mathematics and KlearReasoner-CodeSub-15K (Kwai-Klear, 2025) for code. Following Lightning OPD (Wu et al., 2026), we cache one response per training example from the frozen reference and precompute teacher scores. Training uses slime (Zhu et al., 2025), a global batch size of 256 and runs for 150 steps. We use five prompt-level folds for cross-fitted residualization. Model-specific initializations, learning rates, and training budgets are detailed in Appendix D.

Evaluation. For mathematical reasoning, we evaluate on AIME 2024 (AI-MO, 2024), AIME 2025 (OpenCompass, 2025), and HMMT February 2025 (Balunović et al., 2025). For code generation, we evaluate on LiveCodeBench v5 and v6 (Jain et al., 2025). We use top-$p = 0.95$ and report average Pass@1, with 64 math solutions and 8 code solutions per problem by default. Setting-specific temperatures, generation limits, and evaluation templates are detailed in Appendix D.

Baselines. We compare Lightning OPD 2.0 with the SFT reference and Lightning OPD in all three settings, and additionally include IW-OPD and TA-OPD in the Qwen3-4B-SFT and Klear-Reasoner-8B-SFT settings. Lightning OPD (Wu et al., 2026) applies the original token-level distillation objective without style correction. IW-OPD (Xie et al., 2026) addresses position bias by weighting tokens according to the accumulated teacher–student discrepancy, while TA-OPD (Wang et al., 2026b) selects positions whose teacher signal is predicted to be learnable. Our OPD runs within each setting use the same cached rollouts and teacher log-probabilities. Since IW-OPD and TA-OPD were originally formulated with online rollouts, we adapt their objectives to the frozen Lightning OPD replay, precomputing TA geometry from the frozen reference and selected teacher.

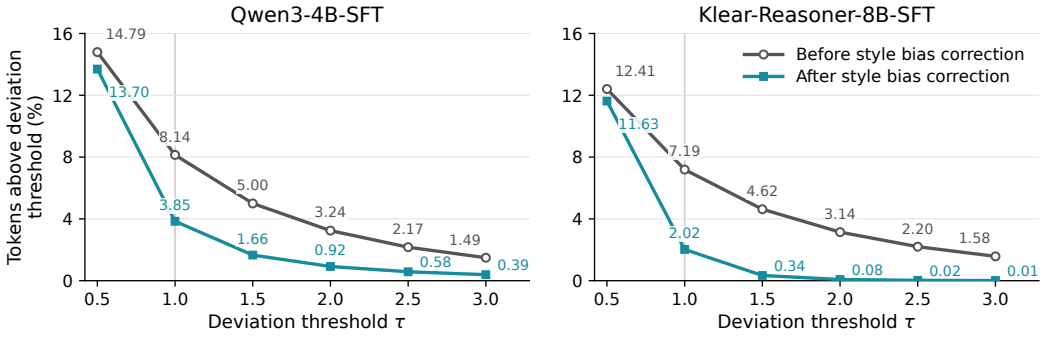


Figure 3: Absolute-deviation threshold analysis. Each curve reports the fraction of scored response tokens whose absolute deviation from the diagnostic reference signal exceeds τ, using o_{it} before style-bias correction and o^*_{it} after correction. The vertical line at $\tau = 1$ marks the representative reference point used in the text. Qwen3-8B provides the exact teacher-consistent reference for Qwen3-4B-SFT, while Klear-Reasoner-8B provides a diagnostic proxy for Klear-Reasoner-8B-SFT. Both models are used only for this post-hoc analysis.

4.2 Main Results

Table 1 shows that Lightning OPD 2.0 improves all five benchmarks over the SFT reference and both domain averages over Lightning OPD in all three settings. In the Qwen3-4B setting, average math and code gains are 3.4 and 3.1 points over SFT and 3.1 and 1.4 points over Lightning OPD. In the Klear-Reasoner-8B-SFT setting, the corresponding gains are 1.0 and 3.6 points over SFT and 1.0 and 1.4 points over Lightning OPD. Among our runs, Lightning OPD 2.0 achieves the strongest domain averages in both settings, exceeding IW-OPD and TA-OPD. In the Olmo-3-7B-Think-SFT setting, Lightning OPD 2.0 remains effective under both teacher and tokenizer mismatch, gaining 2.5 and 1.3 points over SFT and 2.4 and 2.8 points over Lightning OPD on math and code, respectively. These results demonstrate generalization across SFT initializations, data-generation pipelines, and tokenizers without requiring teacher consistency.

4.3 Mechanism Analysis of Style-Token Bias

We assess whether residualization moves the cross-teacher training signal toward the signal obtained under teacher consistency. Let π_D denote the model used for this post-hoc diagnostic and let ℓ^D_{it} be its chosen-token log-probability. For each scored response token in the cache, we measure the absolute deviation of the raw and residualized signals from the diagnostic reference signal

$$o_{it} = \left| d_{it} - \left(\ell^D_{it} - \ell^R_{it}\right) \right|, \qquad o^*_{it} = \left| d^*_{it} - \left(\ell^D_{it} - \ell^R_{it}\right) \right|, \tag{10}$$

where $d_{it} = \ell^T_{it} - \ell^R_{it}$ is the original cross-teacher disagreement and d^*_{it} is its residualized counterpart. In the Qwen3-4B-SFT setting, π_D is Qwen3-8B, the provenance teacher π_G that generated the SFT demonstrations. The diagnostic reference is therefore the exact teacher-consistent signal in this setting. Exact teacher-consistent scores are unavailable in the Klear-Reasoner-8B-SFT setting, so we use Klear-Reasoner-8B (Kwai-Klear, 2025) as a diagnostic proxy. Neither diagnostic model contributes to correction estimation or student training.

Figure 3 compares the empirical fractions of tokens satisfying $o_{it} > \tau$ and $o^*_{it} > \tau$ as τ varies from 0.5 to 3.0. The after-correction curve lies below the before-correction curve at every evaluated threshold in both settings. At $\tau = 1$, residualization reduces the exceedance rate from 8.14% to 3.85% in the Qwen3-4B-SFT setting and from 7.19% to 2.02% in the Klear-Reasoner-8B-SFT setting. Relative reductions grow at higher thresholds.

Figure 4 provides a complementary token-level view using two cached responses from the Qwen3-4B-SFT setting. In these examples, several large raw deviations occur on discourse markers and reasoning transitions. Most fall below the reference threshold after residualization, although some large deviations remain. Together, the aggregate and token-level evidence supports our operational interpretation of the removed component as recurring style-token bias.

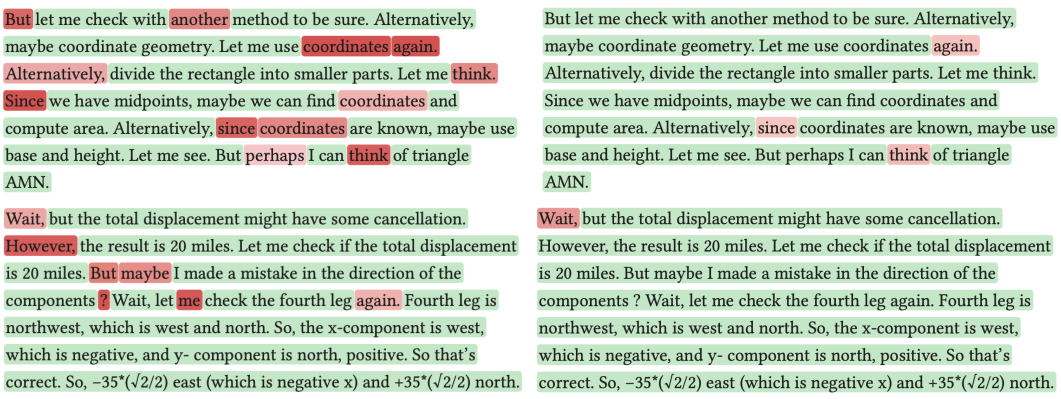


Figure 4: Token-level examples of style-bias correction in the Qwen3-4B-SFT setting. Each row shows one cached response before correction on the left and after residualization on the right. Green denotes an absolute deviation of at most one nat from the diagnostic reference signal. Red denotes a deviation above one nat, with darker shades indicating larger values. The one-nat cutoff is used only for visualization.

Table 2: Component ablations of Lightning OPD 2.0 on AIME 2024 and HMMT February 2025. All entries report Pass@1 in percent. Checkmarks indicate retained components, and boldface marks the highest point estimate in each column.

Variant	Components			Qwen3-4B-SFT		Klear-Reasoner-8B-SFT	
	Token	Context	Cross-fitting	AIME24	HMMT25	AIME24	HMMT25
Lightning OPD	✗	✗	✗	59.6	34.6	80.6	62.9
Context only	✗	✓	✓	61.5	36.6	82.3	62.6
Token only	✓	✗	✓	61.3	35.4	82.1	61.8
In-sample estimation	✓	✓	✗	62.7	**36.7**	81.8	63.1
Lightning OPD 2.0	✓	✓	✓	**63.5**	**36.7**	**82.4**	**63.8**

4.4 Ablation Study

Table 2 ablates the three principal design choices in Lightning OPD 2.0 in the Qwen3-4B-SFT and Klear-Reasoner-8B-SFT settings. The context-only and token-only variants retain one lookup at unit weight, whereas the in-sample variant retains both lookups but removes prompt-level cross-fitting. Lightning OPD serves as the uncorrected baseline, and all variants otherwise share the same cache and training configuration within each setting. The full method achieves the highest or tied-highest score in every column, supporting the combination of lexical and contextual estimates. Cross-fitting matches or improves on in-sample estimation across both settings.

5 Conclusion

Teacher consistency is central to effective OPD, yet it is often unavailable when an SFT reference and a later distillation teacher are selected independently. We study this cross-teacher setting and find that chosen-token disagreement contains potentially useful context-specific teacher evidence together with recurring differences in wording, formatting, and reasoning cadence. Lightning OPD 2.0 estimates this recurring component from other cached rollouts as an operational proxy for style-token bias and subtracts it before the reference-anchored offline OPD update. Across the Qwen3-4B-SFT, Klear-Reasoner-8B-SFT, and Olmo-3-7B-Think-SFT settings, Lightning OPD 2.0 delivers consistent improvements over Lightning OPD on both mathematical reasoning and code generation tasks. Our analyses further validate the proposed design, showing that residualization suppresses the influence of style-token bias on the distillation signal. Together, these results establish Lightning OPD 2.0 as a more general OPD framework, broadening the applicability of on-policy distillation beyond the teacher-consistency constraint.

AI USE STATEMENT

In this work, we used generative AI tools to support qualitative and thematic data analysis. We have not used generative AI tools to generate synthetic data sets beyond the teacher- and student-generated trajectories described in the paper; propose or refine hypotheses; design or provide feedback on research methodology or experiments; implement methods; assist with translation; clean and reformat datasets; or interpret results. The use of generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, or assist in the writing of proofs is not applicable to this work. Additionally, we used generative AI tools to edit the paper to improve readability. The authors reviewed all AI-assisted analyses and language edits. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

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 B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5be69a584901a26c521c2b51e40a4c20-Paper-Conference.pdf.

AI-MO. AIME 2024. https://huggingface.co/datasets/AI-MO/aimo-validation-aime, 2024.

Mohammadreza Armandpour, Fatih Ilhan, David Harrison, Ajay Jaiswal, Duc N. M Hoang, Fartash Faghri, Yizhe Zhang, Minsik Cho, and Mehrdad Farajtabar. Unmasking on-policy distillation: Where it helps, where it hurts, and why, 2026. URL https://arxiv.org/abs/2605.10889.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. MathArena: Evaluating LLMs on uncontaminated math competitions. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-0679. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/1d27c01ebd3e3aebe226b44fc970d803-Paper-Datasets_and_Benchmarks_Track.pdf.

Liang Chen, Xueting Han, Li Shen, Jing Bai, and Kam-Fai Wong. Beyond two-stage training: Cooperative SFT and RL for LLM reasoning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=xWvj03N4sJ.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018. doi: 10.1111/ectj.12097.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=9SkkifLopZ.

DeepSeek-AI. DeepSeek-R1-0528. https://huggingface.co/deepseek-ai/DeepSeek-R1-0528, 2025.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi

Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. DeepSeek-V3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=m7p5O7zblY.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=uZbZb058bA.

GLM-5 Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. GLM-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/8ac015d409635f196f9e3e9dcfb9a94e-Paper-Conference.pdf.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvarna, Benjamin Feuer, Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, Wanjia Zhao, John Yang, Shreyas Pimpalgaonkar, Kartik Sharma, Charlie Cheng-Jie Ji, Yichuan Deng, Sarah Pratt, Vivek Ramanujan, Jon Saad-Falcon, Jeffrey Li, Achal Dave, Alon Albalak, Kushal Arora, Blake Wulfe, Chinmay Hegde, Greg Durrett, Sewoong Oh, Mohit Bansal, Saadia Gabriel, Aditya Grover, Kai-Wei Chang, Vaishaal Shankar, Aaron Gokaslan, Mike A. Merrill, Tatsunori Hashimoto, Yejin Choi, Jenia Jitsev, Reinhard Heckel, Maheswaran Sathiamoorthy, Alexandros G. Dimakis, and Ludwig Schmidt. OpenThoughts: Data recipes for reasoning models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 108059–108130, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/b010241b9f1cdfc7d4c392db899cef86-Paper-Conference.pdf.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, Sep 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

Yongyi Guo, Dominic Coey, Mikael Konutgan, Wenting Li, Chris Schoener, and Matt Goldman. Machine learning for variance reduction in online experiments. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 8637–8648. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/488b084119a1c7a4950f00706ec7ea16-Paper.pdf.

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

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 162239–162262. Curran Associates, Inc., 2025. doi: 10.52202/085713-5418. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/ed873d79e7c268c020c4b4db13a2812a-Paper-Conference.pdf.

Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo Ponti, and Ivan Titov. Blending supervised and reinforcement fine-tuning with prefix sampling. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=MgtyPYrC4h.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=QkfkxyRizZ.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida I. Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 58791–58831, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/94074dd5a072d28ff75a76dabed43767-Paper-Conference.pdf.

Nan Jia, Haojin Yang, Xing Ma, Jiesong Lian, Shuailiang Zhang, Weipeng Zhang, Ke Zeng, Xunliang Cai, and Zequn Sun. Asymmetric on-policy distillation: Bridging exploitation and imitation at the token level, 2026. URL https://arxiv.org/abs/2605.06387.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=J5i09faOOf.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. VinePPO: Refining credit assignment in RL training of LLMs. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 29557–29590. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/kazemnejad25a.html.

Junlong Ke, Zichen Wen, Weijia Li, Conghui He, and Linfeng Zhang. Respecting self-uncertainty in on-policy self-distillation for efficient LLM reasoning, 2026. URL https://arxiv.org/abs/2605.13255.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chuning Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, Haiqing Guo, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haotian Yao, Haotian Zhao, Haoyu Lu, Haoze Li, Haozhen Yu, Hongcheng Gao, Huabin Zheng, Huan Yuan, Jia Chen, Jianhang Guo, Jianlin Su, Jianzhou Wang, Jie Zhao, Jin Zhang, Jingyuan Liu, Junjie Yan, Junyan Wu, Lidong Shi, Ling Ye, Longhui Yu, Mengnan Dong, Neo Zhang, Ningchen Ma, Qiwei Pan, Qucheng Gong, Shaowei Liu, Shengling Ma, Shupeng Wei, Sihan Cao, Siying Huang, Tao Jiang, Weihao Gao, Weimin Xiong, Weiran He, Weixiao Huang, Weixin Xu, Wenhao Wu, Wenyang He, Xianghui Wei, Xianqing Jia, Xingzhe Wu, Xinran Xu, Xinxing Zu, Xinyu Zhou, Xuehai Pan, Y. Charles,

Yang Li, Yangyang Hu, Yangyang Liu, Yanru Chen, Yejie Wang, Yibo Liu, Yidao Qin, Yifeng Liu, Ying Yang, Yiping Bao, Yulun Du, Yuxin Wu, Yuzhi Wang, Zaida Zhou, Zhaoji Wang, Zhaowei Li, Zhen Zhu, Zheng Zhang, Zhexu Wang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Ziyao Xu, Zonghan Yang, and Zongyu Lin. Kimi k1.5: Scaling reinforcement learning with LLMs, 2025. URL https://arxiv.org/abs/2501.12599.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, S. H. Cai, Yuan Cao, Ziwei Chai, Y. Charles, H. S. Che, Cheng Chen, Guanduo Chen, Huarong Chen, Jia Chen, Jianlong Chen, Jun Chen, Kefan Chen, Liang Chen, Ruijue Chen, Xinhao Chen, Yanru Chen, Yanxu Chen, Yicun Chen, Yimin Chen, Yingjiang Chen, Yuankun Chen, Yujie Chen, Yutian Chen, Zhirong Chen, Ziwei Chen, Dazhi Cheng, Yean Cheng, Minghan Chu, Jialei Cui, Jiaqi Deng, Muxi Diao, Hao Ding, Mengfan Dong, Mengnan Dong, Yuxin Dong, Yuhao Dong, Ang'ang Du, Chenzhuang Du, Dikang Du, Lingxiao Du, Yulun Du, Yu Fan, Shengjun Fang, Qiulin Feng, Yichen Feng, Garimugai Fu, Kelin Fu, Hongcheng Gao, Tong Gao, Yuyao Ge, Shangyi Geng, Chengyang Gong, Xiaochen Gong, Zhuoma Gongque, Qizheng Gu, Xinran Gu, Yicheng Gu, Longyu Guan, Shuhao Guan, Yuanying Guo, Xiaoru Hao, Dailan He, Tianhong He, Weiran He, Wenyang He, Yibo He, Yunjia He, Chao Hong, Hao Hu, Jiaxi Hu, Yangyang Hu, Zhenxing Hu, Ke Huang, Ruiyuan Huang, Weixiao Huang, Zhiqi Huang, Chaobo Jia, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yu Jing, Guokun Lai, Aidi Li, C. Li, Cheng Li, Fang Li, Guanghe Li, Guanyu Li, Haitao Li, Haoyang Li, Jia Li, Jingwei Li, Junxiong Li, Lincan Li, Mo Li, Weihong Li, Wentao Li, Xinhang Li, Xinhao Li, Yang Li, Yanhao Li, Yiwei Li, Yuxiao Li, Zhaowei Li, Zhaoxi Li, Zheming Li, Weilong Liao, Jiawei Lin, Xiaohan Lin, Yibo Lin, Zhishan Lin, Zichao Lin, Cheng Liu, Chenyu Liu, Hongzhang Liu, Liang Liu, Shaowei Liu, Shudong Liu, Shuran Liu, Tianwei Liu, Tianyu Liu, Weizhou Liu, Xiangyan Liu, Yangyang Liu, Yanming Liu, Yibo Liu, Yuanxin Liu, Zhengying Liu, Zhongnuo Liu, Enzhe Lu, Haoyu Lu, Zhiyuan Lu, G. Luo, Junyu Luo, Tongxu Luo, Yashuo Luo, Long Ma, Shaoguang Mao, Yuan Mei, Xin Men, Fanqing Meng, Zhiyong Meng, Yibo Miao, Minqing Ni, Kun Ouyang, Siyuan Pan, Bo Pang, Yuchao Qian, Ruoyu Qin, Zeyu Qin, Jiezhong Qiu, Bowen Qu, Zeyu Shang, Youbo Shao, Tianxiao Shen, Zhennan Shen, Juanfeng Shi, Lidong Shi, Shengyuan Shi, Feifan Song, Pengwei Song, Tianhui Song, Xiaoxi Song, Hongjin Su, Jianlin Su, Zhaochen Su, Lin Sui, Jinsong Sun, Junyao Sun, Tongyu Sun, Flood Sung, Yunpeng Tai, Chuning Tang, Heyi Tang, Xiaojuan Tang, Zhengyang Tang, Jiawen Tao, Shiyuan Teng, Chaoran Tian, Pengfei Tian, Bowen Wang, Chensi Wang, Chuang Wang, Congcong Wang, Dingkun Wang, Dinglu Wang, Dongliang Wang, Feng Wang, Hailong Wang, Haiming Wang, Hao Wang, Hengzhi Wang, Huaqing Wang, Hui Wang, Jiahao Wang, Jinhong Wang, Jiuzheng Wang, Kaixin Wang, Linian Wang, Qibin Wang, Shengjie Wang, Shuyi Wang, Si Wang, Wei Wang, Xiaochen Wang, Xinyuan Wang, Yao Wang, Yejie Wang, Yipu Wang, Yiqin Wang, Yucheng Wang, Yuzhi Wang, Zhaoji Wang, Zhaowei Wang, Zhengtao Wang, Zhexu Wang, Zifan Wang, Zihan Wang, Zizhe Wang, Chu Wei, Ming Wei, Chuan Wen, Zichen Wen, Chengjie Wu, Haoning Wu, Junyan Wu, Rucong Wu, Wenhao Wu, Yuefeng Wu, Yuhao Wu, Yuxin Wu, Zijian Wu, Chenjun Xiao, Jin Xie, Xiaotong Xie, Yuchong Xie, Bowei Xing, Boyu Xu, Jianfan Xu, Jing Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinbo Xu, Xinran Xu, Yangchuan Xu, Yichang Xu, Yuemeng Xu, Zelai Xu, Ziyao Xu, Junjie Yan, Yuzi Yan, Guangyao Yang, Hao Yang, Junwei Yang, Kai Yang, Ningyuan Yang, Xiaofei Yang, Xinlong Yang, Xinyu Yang, Ying Yang, Yi Yang, Yi Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Dan Ye, Haoran Ye, Wenjie Ye, Zhuorui Ye, Peng Yebo, Bohong Yin, Chengzhen Yu, Longhui Yu, Tao Yu, Tianxiang Yu, Enming Yuan, Mengjie Yuan, Xiaokun Yuan, Yang Yue, Weihao Zeng, Dunyuan Zha, Haobing Zhan, Dehao Zhang, Hao Zhang, Jin Zhang, Puqi Zhang, Qiao Zhang, Rui Zhang, Xiaobin Zhang, Xiaoyun Zhang, Y. Zhang, Yadong Zhang, Yangkun Zhang, Yichi Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yushun Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Chenguang Zhao, Feifan Zhao, Jinxiang Zhao, Shuai Zhao, Xiangyu Zhao, Xuanle Zhao, Yikai Zhao, Zijia Zhao, Huabin Zheng, Ruihan Zheng, Shaojie Zheng, Tengyang Zheng, Junfeng Zhong, Longguang Zhong, Weiming Zhong, M. Zhou, Runjie Zhou, Xinyu Zhou, Zaida Zhou, Jinguo Zhu, Liya Zhu, Xinhao Zhu, Yuxuan Zhu, Zhen Zhu, Jingze Zhuang, Weiyu Zhuang, Ying Zou, and Xinxing Zu. Kimi k2.5: Visual agentic intelligence, 2026. URL https://arxiv.org/abs/2602.02276.

Kwai-Klear. Klear-Reasoner-8B-SFT. https://huggingface.co/Kwai-Klear/Klear-Reasoner-8B-SFT, 2025.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation

of large language models: Phenomenology, mechanism, and recipe. In *ICML 2026 Workshop on Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning*, 2026. URL https://openreview.net/forum?id=aaDDXXDGXB.

Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. ReMax: A simple, effective, and efficient reinforcement learning method for aligning large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 29128–29163. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/li24cd.html.

Kun Liang, Clive Bai, Xin Xu, Chenming Tang, Sanwoo Lee, Weijie Liu, Saiyong Yang, and Yunfang Wu. ORBIT: On-policy exploration-exploitation for controllable multi-budget reasoning, 2026. URL https://arxiv.org/abs/2601.08310.

Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. UFT: Unifying supervised and reinforcement fine-tuning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 101347–101383. Curran Associates, Inc., 2025a. doi: 10.52202/085713-3394. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/92e886487a8354b03d8bf4416eae6d7d-Paper-Conference.pdf.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025b. URL https://openreview.net/forum?id=5PAF7PAY2Y.

LLM-Core Xiaomi, Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, Gang Xie, Hailin Zhang, Hanglong Lv, Hanyu Li, Heyu Chen, Hongshen Xu, Houbin Zhang, Huaqiu Liu, Jiangshan Duo, Jianyu Wei, Jiebao Xiao, Jinhao Dong, Jun Shi, Junhao Hu, Kainan Bao, Kang Zhou, Lei Li, Liang Zhao, Linghao Zhang, Peidian Li, Qianli Chen, Shaohui Liu, Shihua Yu, Shijie Cao, Shimao Chen, Shouqiu Yu, Shuo Liu, Tianling Zhou, Weijiang Su, Weikun Wang, Wenhan Ma, Xiangwei Deng, Xing Zhang, Yifan Song, Yihan Yan, Yihao Zhao, Yingchun Lai, Yizhao Gao, Yu Cheng, Yuanyuan Tian, Yudong Wang, Zhen Tang, Zhengju Tang, Zhengtao Wen, Zhichao Song, Zhixian Zheng, Zihan Jiang, Bohan Mao, Bowen Ye, Can Cai, Chenghua Wang, Chengxuan Zhu, Chong Ma, Chun Chen, Chunan Li, Dawei Zhu, Deshan Xiao, Dong Zhang, Duo Zhang, Fangyue Liu, Feiyu Yang, Fengyuan Shi, Guoan Wang, Hao Tian, Hao Wu, Heng Qu, Hongfei Yi, Hongxu An, Hongyi Guan, Jian Wen, Jiarui Sun, Jiawei Li, Jinlong Xue, Jun Xia, Kai Fang, Menghang Zhu, Nuo Chen, Qian Tu, Qihao Zhang, Qiying Wang, Rang Li, Rui Ma, Shaolei Zhang, Shengfan Wang, Shicheng Li, Shuhao Gu, Shuhuai Ren, Sirui Deng, Tao Guo, Tianyang Lu, Weiji Zhuang, Weikang Zhang, Weimin Xiong, Wenshan Huang, Wenyu Yang, Xin Zhang, Xing Yong, Xu Wang, Xueyang Xie, Yilin Jiang, Yixin Yang, Yongzhe He, Yu Tu, Yuanliang Dong, Yuchen Liu, Yue Ma, Yue Yu, Yuxing Xiang, Zhaojun Huang, Zhenru Lin, Zhipeng Xu, Zhiyang Chen, Zhonghua Deng, Zihan Zhang, and Zihao Yue. MiMo-V2-Flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, Bin Cui, and Wentao Zhang. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 80802–80821, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/82c4ad0c3d3d4898a77be5df7b2ede3a-Paper-Conference.pdf.

MiniMax, Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, Chengjun Xiao, Chengyu Du, Chi Zhang, Chu Qiao, Chunhao Zhang, Chunhui Du, Congchao Guo, Da Chen, Deming Ding, Dianjun Sun, Dong Li, Enwei Jiao, Haigang Zhou, Haimo Zhang, Han Ding, Haohai Sun, Haoyu Feng, Huaiguang Cai, Haichao Zhu, Jian Sun, Jiaqi Zhuang, Jiaren Cai, Jiayuan Song, Jin Zhu, Jingyang Li, Jinhao

Tian, Jinli Liu, Junhao Xu, Junjie Yan, Junteng Liu, Junxian He, Kaiyi Feng, Ke Yang, Kecheng Xiao, Le Han, Leyang Wang, Lianfei Yu, Liheng Feng, Lin Li, Lin Zheng, Linge Du, Lingyu Yang, Lunbin Zeng, Minghui Yu, Mingliang Tao, Mingyuan Chi, Mozhi Zhang, Mujie Lin, Nan Hu, Nongyu Di, Peng Gao, Pengfei Li, Pengyu Zhao, Qibing Ren, Qidi Xu, Qile Li, Qin Wang, Rong Tian, Ruitao Leng, Shaoxiang Chen, Shaoyu Chen, Shengmin Shi, Shitong Weng, Shuchang Guan, Shuqi Yu, Sichen Li, Songquan Zhu, Tengfei Li, Tianchi Cai, Tianrun Liang, Weiyu Cheng, Weize Kong, Wenkai Li, Xiancai Chen, Xiangjun Song, Xiao Luo, Xiao Su, Xiaobo Li, Xiaodong Han, Xinzhu Hou, Xuan Lu, Xun Zou, Xuyang Shen, Yan Gong, Yan Ma, Yang Wang, Yiqi Shi, Yiran Zhong, Yonghong Duan, Yongxiang Fu, Yongyi Hu, Yu Gao, Yuanxiang Fan, Yufeng Yang, Yuhao Li, Yulin Hu, Yunan Huang, Yunji Li, Yunzhi Xu, Yuxin Mao, Yuxuan Shi, Yuze Wenren, Zehan Li, Zelin Li, Zhanxu Tian, Zhengmao Zhu, Zhenhua Fan, Zhenzhen Wu, Zhichao Xu, Zhihang Yu, Zhiheng Lyu, Zhuo Jiang, Zibo Gao, Zijia Wu, Zijian Song, and Zijun Sun. MiniMax-M1: Scaling test-time compute efficiently with lightning attention, 2025. URL https://arxiv.org/abs/2506.13585.

NVIDIA, Aaron Blakeman, Aaron Grattafiori, Aarti Basant, Abhibha Gupta, Abhinav Khattar, Adi Renduchintala, Aditya Vavre, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, Aleksandr Shaposhnikov, Alex Kondratenko, Alexander Bukharin, Alexandre Milesi, Ali Taghibakhshi, Alisa Liu, Amelia Barton, Ameya Sunil Mahabaleshwarkar, Amir Klein, Amit Zuker, Amnon Geifman, Amy Shen, Anahita Bhiwandiwalla, Andrew Tao, Anjulie Agrusa, Ankur Verma, Ann Guan, Anubhav Mandarwal, Arham Mehta, Ashwath Aithal, Ashwin Poojary, Asif Ahamed, Asit Mishra, Asma Kuriparambil Thekkumpate, Ayush Dattagupta, Banghua Zhu, Bardiya Sadeghi, Barnaby Simkin, Ben Lanir, Benedikt Schifferer, Besmira Nushi, Bilal Kartal, Bita Darvish Rouhani, Boris Ginsburg, Brandon Norick, Brandon Soubasis, Branislav Kisacanin, Brian Yu, Bryan Catanzaro, Carlo del Mundo, Chantal Hwang, Charles Wang, Cheng-Ping Hsieh, Chenghao Zhang, Chenhan Yu, Chetan Mungekar, Chintan Patel, Chris Alexiuk, Christopher Parisien, Collin Neale, Cyril Meurillon, Damon Mosk-Aoyama, Dan Su, Dane Corneil, Daniel Afrimi, Daniel Lo, Daniel Rohrer, Daniel Serebrenik, Daria Gitman, Daria Levy, Darko Stosic, David Mosallanezhad, Deepak Narayanan, Dhruv Nathawani, Dima Rekesh, Dina Yared, Divyanshu Kakwani, Dong Ahn, Duncan Riach, Dusan Stosic, Edgar Minasyan, Edward Lin, Eileen Long, Eileen Peters Long, Elad Segal, Elena Lantz, Ellie Evans, Elliott Ning, Eric Chung, Eric Harper, Eric Tramel, Erick Galinkin, Erik Pounds, Evan Briones, Evelina Bakhturina, Evgeny Tsykunov, Faisal Ladhak, Fay Wang, Fei Jia, Felipe Soares, Feng Chen, Ferenc Galko, Frank Sun, Frankie Siino, Gal Hubara Agam, Ganesh Ajjanagadde, Gantavya Bhatt, Gargi Prasad, George Armstrong, Gerald Shen, Gorkem Batmaz, Grigor Nalbandyan, Haifeng Qian, Harsh Sharma, Hayley Ross, Helen Ngo, Herbert Hum, Herman Sahota, Hexin Wang, Himanshu Soni, Hiren Upadhyay, Huizi Mao, Huy C Nguyen, Huy Q Nguyen, Iain Cunningham, Ido Galil, Ido Shahaf, Igor Gitman, Ilya Loshchilov, Itamar Schen, Itay Levy, Ivan Moshkov, Izik Golan, Izzy Putterman, Jan Kautz, Jane Polak Scowcroft, Jared Casper, Jatin Mitra, Jeffrey Glick, Jenny Chen, Jesse Oliver, Jian Zhang, Jiaqi Zeng, Jie Lou, Jimmy Zhang, Jinhang Choi, Jining Huang, Joey Conway, Joey Guman, John Kamalu, Johnny Greco, Jonathan Cohen, Joseph Jennings, Joyjit Daw, Julien Veron Vialard, Junkeun Yi, Jupinder Parmar, Kai Xu, Kan Zhu, Kari Briski, Katherine Cheung, Katherine Luna, Keith Wyss, Keshav Santhanam, Kevin Shih, Kezhi Kong, Khushi Bhardwaj, Kirthi Shankar, Krishna C. Puvvada, Krzysztof Pawelec, Kumar Anik, Lawrence McAfee, Laya Sleiman, Leon Derczynski, Li Ding, Lizzie Wei, Lucas Liebenwein, Luis Vega, Maanu Grover, Maarten Van Segbroeck, Maer Rodrigues de Melo, Mahdi Nazemi, Makesh Narsimhan Sreedhar, Manoj Kilaru, Maor Ashkenazi, Marc Romeijn, Marcin Chochowski, Mark Cai, Markus Kliegl, Maryam Moosaei, Matt Kulka, Matvei Novikov, Mehrzad Samadi, Melissa Corpuz, Mengru Wang, Meredith Price, Michael Andersch, Michael Boone, Michael Evans, Miguel Martinez, Mikail Khona, Mike Chrzanowski, Minseok Lee, Mohammad Dabbah, Mohammad Shoeybi, Mostofa Patwary, Nabin Mulepati, Najeeb Nabwani, Natalie Hereth, Nave Assaf, Negar Habibi, Neta Zmora, Netanel Haber, Nicola Sessions, Nidhi Bhatia, Nikhil Jukar, Nikki Pope, Nikolai Ludwig, Nima Tajbakhsh, Nir Ailon, Nirmal Juluru, Nishant Sharma, Oleksii Hrinchuk, Oleksii Kuchaiev, Olivier Delalleau, Oluwatobi Olabiyi, Omer Ullman Argov, Omri Puny, Oren Tropp, Ouye Xie, Parth Chadha, Pasha Shamis, Paul Gibbons, Pavlo Molchanov, Pawel Morkisz, Peter Dykas, Peter Jin, Pinky Xu, Piotr Januszewski, Pranav Prashant Thombre, Prasoon Varshney, Pritam Gundecha, Przemek Tredak, Qing Miao, Qiyu Wan, Rabeeh Karimi Mahabadi, Rachit Garg, Ran El-Yaniv, Ran Zilberstein, Rasoul Shafipour, Rich Harang, Rick Izzo, Rima Shahbazyan, Rishabh Garg, Ritika Borkar, Ritu Gala, Riyad Islam, Robert Hesse, Roger Waleffe, Rohit Watve, Roi Koren, Ruoxi Zhang, Russell Hewett, Russell J. Hewett, Ryan

Prenger, Ryan Timbrook, Sadegh Mahdavi, Sahil Modi, Samuel Kriman, Sangkug Lim, Sanjay Kariyappa, Sanjeev Satheesh, Saori Kaji, Satish Pasumarthi, Saurav Muralidharan, Sean Narentharen, Sean Narenthiran, Seonmyeong Bak, Sergey Kashirsky, Seth Poulos, Shahar Mor, Shanmugam Ramasamy, Shantanu Acharya, Shaona Ghosh, Sharath Turuvekere Sreenivas, Shelby Thomas, Shiqing Fan, Shreya Gopal, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shuoyang Ding, Siddharth Singh, Simeng Sun, Smita Ithape, Somshubra Majumdar, Soumye Singhal, Stas Sergienko, Stefania Alborghetti, Stephen Ge, Sugam Dipak Devare, Sumeet Kumar Barua, Suseella Panguluri, Suyog Gupta, Sweta Priyadarshi, Syeda Nahida Akter, Tan Bui, Teodor-Dumitru Ene, Terry Kong, Thanh Do, Tijmen Blankevoort, Tim Moon, Tom Balough, Tomer Asida, Tomer Bar Natan, Tomer Ronen, Tugrul Konuk, Twinkle Vashishth, Udi Karpas, Ushnish De, Vahid Noorozi, Vahid Noroozi, Venkat Srinivasan, Venmugil Elango, Victor Cui, Vijay Korthikanti, Vinay Rao, Vitaly Kurin, Vitaly Lavrukhin, Vladimir Anisimov, Wanli Jiang, Wasi Uddin Ahmad, Wei Du, Wei Ping, Wenfei Zhou, Will Jennings, William Zhang, Wojciech Prazuch, Xiaowei Ren, Yashaswi Karnati, Yejin Choi, Yev Meyer, Yi-Fu Wu, Yian Zhang, Yigong Qin, Ying Lin, Yonatan Geifman, Yonggan Fu, Yoshi Subara, Yoshi Suhara, Yubo Gao, Zach Moshe, Zhen Dong, Zhongbo Zhu, Zihan Liu, Zijia Chen, and Zijie Yan. NVIDIA nemotron 3: Efficient and open intelligence, 2025. URL https://arxiv.org/abs/2512.20856.

Olmo Team, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2026. URL https://arxiv.org/abs/2512.13961.

OpenCompass. AIME 2025. https://github.com/open-compass/opencompass, 2025.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. 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 Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ebZcMQImhG.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023. doi: 10.52202/075280-2338. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.

Miao Rang, Zhenni Bi, Hang Zhou, Hanting Chen, An Xiao, Tianyu Guo, Kai Han, Xinghao Chen, and Yunhe Wang. Revealing the power of post-training for small language models via knowledge distillation, 2025. URL https://arxiv.org/abs/2509.26497.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=qA6FgH0nnZ.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL's razor: Why online reinforcement learning forgets less. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=7HNRYT4V44.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models, 2026. URL https://arxiv.org/abs/2604.00626.

Zhiquan Tan and Yinrong Hong. Self-supervised on-policy distillation for reasoning language models, 2026. URL https://arxiv.org/abs/2605.17497.

Boxin Wang, Chankyu Lee, Nayeon Lee, Sheng-Chieh Lin, Wenliang Dai, Yang Chen, Yangyi Chen, Zhuolin Yang, Zihan Liu, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nemotron-Cascade: Scaling cascaded reinforcement learning for general-purpose reasoning models, 2026a. URL https://arxiv.org/abs/2512.13607.

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

Ethan G. Wilcox, Tiago Pimentel, Clara Meister, Ryan Cotterell, and Roger P. Levy. Testing the predictions of surprisal theory in 11 languages. *Transactions of the Association for Computational Linguistics*, 11:1451–1470, 2023. doi: 10.1162/tacl_a_00612. URL https://aclanthology.org/2023.tacl-1.82/.

Yecheng Wu, Song Han, and Han Cai. Lightning OPD: Efficient post-training for large reasoning models with offline on-policy distillation, 2026. URL https://arxiv.org/abs/2604.13010.

Yan Xie, Sijie Zhu, Tiansheng Wen, Bo Chen, and Yifei Wang. On the position bias of on-policy distillation, 2026. URL https://arxiv.org/abs/2606.22600.

Haoran Xu, Hongyu Wang, Yifei Gao, Jiaze Li, Xiaofeng Zhang, and Xiaosong Yuan. SG-OPD: Sign-gated on-policy distillation via sign-consistency gating and phased teacher sampling, 2026. URL https://arxiv.org/abs/2606.09304.

Wenda Xu, Rujun Han, Zifeng Wang, Long T. Le, Dhruv Madeka, Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 64616–64646, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a2747a3844ca1e4667fbff3f558eb39b-Paper-Conference.pdf.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 117157–117186. Curran Associates, Inc., 2025. doi: 10.52202/085713-3904. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a9d5c33e38442450f9781bd7bad3c81e-Paper-Conference.pdf.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-Math technical report: Toward mathematical expert model via self-improvement, 2024. URL https://arxiv.org/abs/2409.12122.

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, 2025. URL https://arxiv.org/abs/2505.09388.

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

Yuxiao Yang, Xiaoyun Wang, and Weitong Zhang. OGLS-SD: On-policy self-distillation with outcome-guided logit steering for LLM reasoning, 2026b. URL https://arxiv.org/abs/2605.12400.

Zhuolin Yang, Zihan Liu, Yang Chen, Wenliang Dai, Boxin Wang, Sheng-Chieh Lin, Chankyu Lee, Yangyi Chen, Dongfu Jiang, Jiafan He, Renjie Pi, Grace Lam, Nayeon Lee, Alexander Bukharin, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nemotron-Cascade 2: Post-training LLMs with cascade RL and multi-domain on-policy distillation, 2026c. URL https://arxiv.org/abs/2603.19220.

Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-box on-policy distillation of large language models, 2026. URL https://arxiv.org/abs/2511.10643.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 113222–113244. Curran Associates, Inc., 2025. doi: 10.52202/085713-3775. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a4277440d50f1f15d2cb4c14f7e0c0d2-Paper-Conference.pdf.

Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. What's behind PPO's collapse in long-CoT? value optimization holds the secret, 2025. URL https://arxiv.org/abs/2503.01491.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 57654–57689. Curran Associates, Inc., 2025a. doi: 10.52202/085713-1933. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/537d5aa768c2d534016a4d06f87bc8fb-Paper-Conference.pdf.

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, Xiangpeng Wei, Xiangyu Yu, Gaohong Liu, Juncai Liu, Lingjun Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Ru Zhang, Xin Liu, Mingxuan Wang, Yonghui Wu, and Lin Yan. VAPO: Efficient and reliable reinforcement learning for advanced reasoning tasks, 2025b. URL https://arxiv.org/abs/2504.05118.

Wenhao Zhang, Yuexiang Xie, Yuchang Sun, Yanxi Chen, Guoyin Wang, Yaliang Li, Bolin Ding, and Jingren Zhou. On-policy RL meets off-policy experts: Harmonizing supervised fine-tuning and reinforcement learning via dynamic weighting. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 120693–120726,

2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/c44a04289beaf0a7d968a94066a1d696-Paper-Conference.pdf.

Zhaoyang Zhang, Shuli Jiang, Yantao Shen, Yuting Zhang, Dhananjay Ram, Shuo Yang, Zhuowen Tu, Wei Xia, and Stefano Soatto. Reinforcement-aware knowledge distillation for LLM reasoning. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=0RBMhFule7.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Jpxfof0EaS.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.

Zilin Zhu, Chengxing Xie, Xin Lv, and slime Contributors. slime: An LLM post-training framework for RL scaling. https://github.com/THUDM/slime, 2025. GitHub repository.

A Cache Construction and Tokenizer Alignment

A.1 Cache Construction

Each cached row contains a prompt, a frozen response, response token IDs, an active-token mask, and aligned chosen-token log-probabilities from the OPD teacher and the SFT reference. We deterministically partition prompts into five folds and keep all responses associated with the same prompt in one fold. The Qwen3-4B-SFT and Klear-Reasoner-8B-SFT settings use compatible Qwen-family tokenization. The Olmo setting uses the cross-tokenizer alignment below, with both scores stored at the original student response positions.

A.2 Cross-Tokenizer Score Alignment

Student-native trajectories. The Olmo–Qwen setting is both cross-teacher and cross-tokenizer. We retain Olmo's native tokenizer and chat template during rollout collection and student training. For scoring, we reconstruct the prompt from the original messages using the student template and preserve the cached response token IDs, including terminal special tokens. The teacher therefore scores the same serialized prompt and response that the student observed.

Byte-preserving projection. Both tokenizers use byte-level BPE. Define $\Phi(y) = (v_1, \ldots, v_m)$ as a nonempty teacher-token path for student token y. A vocabulary symbol shared by the two tokenizers maps directly to its teacher token ID. For other regular symbols, we apply the teacher's BPE model to the student's byte-level vocabulary symbol. This preserves the underlying bytes even when an individual token spans an incomplete UTF-8 character. Nonshared added tokens are encoded from their literal strings and checked for exact decoding round trips. We extend Φ to sequences by concatenating the per-token paths and verify that the projected prompt and response preserve the original serialized content.

Scoring and aggregation. Let q_i^S denote the student-tokenized prompt and write $\Phi(y_{it}) = (v_{it1}, \ldots, v_{itm_{it}})$. The score assigned to student response token y_{it} is

$$\ell_{it}^T = \sum_{j=1}^{m_{it}} \log \pi_T\big(v_{itj} \mid \Phi(q_i^S), \Phi(y_{i,<t}), v_{it,<j}\big)\,. \tag{11}$$

Each conditional probability is normalized over the teacher's full output vocabulary at temperature 1. We sum all conditional log-probabilities along the specified forced path without length normalization. Tokens that expand to multiple teacher tokens retain their original student loss mask. Reference scores ℓ_{it}^R and current-policy scores remain native student-token log-probabilities. Thus, token identities, response positions, reference surprisal, and training losses all use student coordinates, and the cross-fitted estimator and policy update apply unchanged. The aligned teacher and reference scores are cached once before training.

B Cross-Fitted Group Estimator

Let $\mathcal{A}_i \subseteq \{1, \ldots, L_i\}$ be the active token positions in response i and let $n_i = |\mathcal{A}_i|$. For N cached responses, the response-balanced finite-cache measure is

$$Q(i,t) = \frac{1}{N}\frac{1}{n_i}, \qquad i \in \{1, \ldots, N\},\ t \in \mathcal{A}_i. \tag{12}$$

The group estimator uses the corresponding unnormalized response-equivalent weight $NQ(i,t) = 1/n_i$. This scale allows the smoothing prior below to be expressed in response equivalents. We use 32 position bins, 32 reference-surprisal bins, and a reference-surprisal cap of 12.

For numerical stabilization in the implementation, define the symmetric clipping operator

$$C_c(a) = \min\{c, \max\{-c, a\}\}. \tag{13}$$

Let $\mathcal{I}_k$ be the set of rollout indices assigned to fold k. The implementation fits the lookup tables to the bounded score $u_{it} = C_8(d_{it})$, where the raw disagreement d_{it} is defined in Equation (3).

For an estimation group g in coordinate family $j \in \{\mathrm{tok}, \mathrm{ctx}\}$, the non-held-out response mass and score sum are

$$W_j^{(-k)}(g) = \sum_{i \notin \mathcal{I}_k} \sum_{t \in \mathcal{A}_i} \frac{\mathbf{1}[z_{it}^j = g]}{n_i}, \qquad S_j^{(-k)}(g) = \sum_{i \notin \mathcal{I}_k} \sum_{t \in \mathcal{A}_i} \frac{u_{it} \mathbf{1}[z_{it}^j = g]}{n_i}. \tag{14}$$

Let $\mu^{(-k)}$ be the corresponding global response-balanced mean of u. The empirical-Bayes group estimate is

$$m_j^{(-k)}(g) = \frac{S_j^{(-k)}(g) + \eta_j \mu^{(-k)}}{W_j^{(-k)}(g) + \eta_j}, \tag{15}$$

with prior masses $\eta_{\mathrm{tok}} = 0.25$ and $\eta_{\mathrm{ctx}} = 2.0$ in response equivalents. An unseen group therefore backs off to $\mu^{(-k)}$ rather than receiving an undefined or zero estimate.

Applying the lookup structure in Equation (6) to u gives the held-out prediction $\widehat{b}_{it}^{\mathrm{impl}}$. The target builder then forms

$$\delta_{it}^{\mathrm{base}} = C_1(d_{it}), \qquad \bar{b}_{it} = C_1(\widehat{b}_{it}^{\mathrm{impl}}), \qquad \delta_{it}^{\mathrm{res}} = C_1(\delta_{it}^{\mathrm{base}} - \bar{b}_{it}). \tag{16}$$

The implemented effective teacher score is $\widetilde{\ell}_{it}^{T,\mathrm{impl}} = \ell_{it}^R + \delta_{it}^{\mathrm{res}}$, and the corresponding advantage is

$$A_{it}^{\mathrm{impl}}(\theta) = \delta_{it}^{\mathrm{res}} + \ell_{it}^R - \log \pi_\theta(y_{it} \mid h_{it}). \tag{17}$$

This bounded implementation is used only for numerical stabilization. The conceptual method remains the raw residualization in Equation (8). Keeping the fit and objective clips separate lets the estimator observe the magnitude of systematic outliers without allowing those outliers to dominate the training coefficient.

Training uses the default sample-level loss reduction in which active token losses are averaged within each response and then averaged across the global batch. This matches the response-balanced measure Q in Equation (12). Klear adds low-variance reference KL with coefficient 0.01. Matched Klear controls use the same domain-specific term. For Lightning OPD 2.0 and Lightning OPD, the Qwen3-4B-SFT and Olmo-3-7B-Think-SFT settings use a coefficient of zero.

C Training Objective

For each frozen response in an optimization batch, the actor first supplies an old-policy chosen-token log-probability ℓ_{it}^{old}; this is distinct from both the cached teacher score ℓ^T and the reference score ℓ^R. Define

$$\rho_{it}(\theta) = \exp\big(\log \pi_\theta(y_{it} \mid h_{it}) - \ell_{it}^{\mathrm{old}}\big). \tag{18}$$

With the detached implementation advantage from Equation (17), the policy loss is

$$\mathcal{L}_{\mathrm{PG}}(\theta) = \mathbb{E}_Q\Big[\max\Big\{-\rho_{it}(\theta) A_{it}^{\mathrm{impl}}, -\operatorname{clip}(\rho_{it}(\theta), 1-\epsilon_{\mathrm{PPO}}, 1+\epsilon_{\mathrm{PPO}})\, A_{it}^{\mathrm{impl}}\Big\}\Big], \tag{19}$$

where $\epsilon_{\mathrm{PPO}} = 0.2$ on both sides. At $\rho_{it} = 1$, the negative gradient of the unclipped branch recovers the advantage-weighted policy-gradient update corresponding to Equation (17).

Klear adds $\beta_{\mathrm{KL}} \mathbb{E}_Q[k_{it}(\theta)]$ with $\beta_{\mathrm{KL}} = 0.01$, using the low-variance sampled-token estimator

$$k_{it}(\theta) = \operatorname{clip}(\exp(a_{it}) - 1 - a_{it}, -10, 10), \qquad a_{it} = \ell_{it}^R - \log \pi_\theta(y_{it} \mid h_{it}). \tag{20}$$

For Lightning OPD 2.0 and Lightning OPD, Qwen and Olmo set $\beta_{\mathrm{KL}} = 0$. The entropy coefficient is zero. Sequence-level importance reweighting of the frozen rollouts, token rejection, off-policy sequence masking, dual clipping, and all optional auxiliary losses are disabled in these runs.

D Training and Evaluation Settings

D.1 Models and Frozen Trajectories

We study Qwen3-4B-SFT, Klear-Reasoner-8B-SFT, and Olmo-3-7B-Think-SFT (Olmo Team et al., 2026) with Qwen3-30B-A3B-Thinking-2507 (Yang et al., 2025) as the selected OPD teacher. Math

caches contain 17,398 responses from DAPO-Math-17k (Yu et al., 2025); code training uses 15,001 cached responses from KlearReasoner-CodeSub-15K (Kwai-Klear, 2025). We freeze one response per source row and cache aligned chosen-token teacher and reference scores. All responses sharing a prompt are assigned to the same cross-fitting fold. Within each setting, our OPD variants share the frozen rollouts and teacher scores.

The cache representation and cross-tokenizer scoring procedure are described in Appendix A. Qwen and Klear use compatible Qwen-family tokenization, while Olmo retains its native tokenizer and chat template. The cross-tokenizer projection preserves all student response tokens and their loss masks.

D.2 Optimization

Training uses a global batch size of 256 and the residualization parameters in Appendix B. In the main experiments, math runs start from the SFT references and use 150 updates for Qwen, Klear, and Olmo models. For code, we start from math-trained checkpoints and train for 150 further updates. Tables 3–5 summarize the key training hyperparameters. IW-OPD and TA-OPD share the corresponding frozen replay and teacher scores. Training uses BF16 parameters and FP32 gradient reductions.

Table 3: Qwen3-4B-SFT training hyperparameters.

Hyperparameter	**Value**	**Hyperparameter**	**Value**
Optimizer updates	150	Global batch size	256
Learning rate	2×10^{-6}	LR schedule	Constant
Maximum response length	4,096	Adam (β_1, β_2)	$(0.9, 0.98)$
Weight decay	0.1	Gradient clipping	1.0
PPO clip	0.2	Reference KL coefficient	0

Table 4: Default Klear-8B-SFT training hyperparameters.

Hyperparameter	**Value**	**Hyperparameter**	**Value**
Optimizer updates	150	Global batch size	256
Learning rate	2×10^{-7}	LR schedule	Constant
Maximum response length	4,096	Adam (β_1, β_2)	$(0.9, 0.98)$
Weight decay	0.1	Gradient clipping	1.0
PPO clip	0.2	Reference KL coefficient	0.01

Table 5: Olmo-3-7B-Think-SFT training hyperparameters.

Hyperparameter	**Value**	**Hyperparameter**	**Value**
Optimizer updates	150	Global batch size	256
Learning rate	2×10^{-6}	LR schedule	Constant
Maximum response length	4,096	Adam (β_1, β_2)	$(0.9, 0.98)$
Weight decay	0.1	Gradient clipping	1.0
PPO clip	0.2	Reference KL coefficient	0

D.3 Evaluation

Math evaluation covers AIME 2024, AIME 2025, and HMMT February 2025, each with 30 problems. Code evaluation uses LiveCodeBench v5 and v6, with 279 and 131 problems, respectively. We use 64 math generations and eight code generations per problem by default. Domain averages are unweighted means over the three math benchmarks or two code benchmarks.

Local decoding uses top-$p = 0.95$ and temperature 0.6. The math generation limit is 32,768 tokens for Qwen and 40,960 for Klear and Olmo; code evaluations use a limit of 40,960 tokens. Evaluation context limits are 40,960 tokens for Qwen, 81,920 for Klear, and 49,152 for Olmo.

D.4 Evaluation Prompt Templates

We use the following prompt templates for evaluation.

Mathematics prompt. Before application of the model's native chat template, each mathematics example is represented exactly as

```
Question: {Problem}
Answer:
```

LiveCodeBench prompt. The `CodeQwenInstruct` user message begins with:

```
You will be given a question (problem specification) and
will generate a correct Python program that matches the
specification and passes all tests. You will NOT return
anything except for the program.
Question: {question_content}
```

For problems with starter code, the message then asks the model to complete the supplied code and enclose the answer in code delimiters. Otherwise it instructs the model to read from standard input, write to standard output, and return the program inside a Python code block initialized with `# YOUR CODE HERE`. The system message is `You are a helpful assistant.`, wrapped in the Qwen instruction delimiters.

E Additional Experiments

E.1 Global Affine Control

To assess whether global score calibration accounts for the gains of Lightning OPD 2.0, we compare it with a moment-matched affine control:

$$\delta_{it}^{\mathrm{aff}} = C_1(\alpha C_1(d_{it}) + \beta)\,. \tag{21}$$

For each student, we fit the scalar parameters α and β on the frozen training cache to match the mean and second moment of $\delta_{it}^{\mathrm{res}}$ from Equation (16), weighting all active cached tokens equally. The control applies the same global transformation at every position, without token- or context-specific correction.

Both variants use the same frozen trajectories, clipping, and optimization settings within each student. This ablation uses 150 updates for Qwen and Klear. We evaluate on AIME 2024 with 64 generations per problem, temperature 0.6, and top-$p = 0.95$. Table 6 shows that Lightning OPD 2.0 improves over the global affine control by 1.2 and 1.1 percentage points, respectively, supporting the value of conditional correction beyond matching aggregate score moments.

Table 6: Moment-matched global affine control on AIME 2024. All entries report Pass@1 in percent with 64 generations per problem. Boldface marks the highest point estimate in each column.

Method	Qwen3-4B-SFT	Klear-Reasoner-8B-SFT
Global affine control	62.3	81.3
Lightning OPD 2.0	**63.5**	**82.4**

E.2 Hyperparameter Sensitivity

We examine the sensitivity of Lightning OPD 2.0 to the position and reference-surprisal bin counts, smoothing prior masses, number of cross-fitting folds, and objective clipping threshold. Using Qwen3-4B-SFT with Qwen3-30B-A3B-Thinking-2507, we vary one parameter category at a time. The default uses five cross-fitting folds and the estimator settings in Appendix B. The two bin counts and the two prior masses are varied jointly within their respective categories. When varying the objective clip, we apply the same threshold in cache construction and the training objective.

All configurations in Table 7, including the default, start from the same SFT reference and use the same frozen trajectories for 150 optimizer updates. Evaluation uses AIME 2024, temperature 0.6, and top-$p = 0.95$. Varying the bin counts or cross-fitting folds yields scores between 60.8% and 62.1%, compared with 63.5% for the default. The default smoothing prior masses and objective clipping threshold give the highest scores within their respective sweeps.

Table 7: Hyperparameter sensitivity of Lightning OPD 2.0 with Qwen3-4B-SFT. Each configuration uses 150 optimizer updates. AIME24 denotes Pass@1 score.

Parameter category	**Setting**	**AIME24**
Default	$K = 5$; Appendix B	**63.5**
Position / surprisal bins	$(16, 16)$	62.1
	$(64, 64)$	61.9
Token / context prior masses	$(0.125, 1)$	60.8
	$(0.5, 4)$	55.4
Cross-fitting folds	2	60.8
	10	61.7
Objective clip	0.5	58.3
	2	61.3

E.3 Additional Teacher Models

We evaluate Polaris-4B-Preview as an additional OPD teacher for Qwen3-4B-SFT. All distilled models start from the same SFT reference and use the same frozen math trajectories for 150 optimizer updates, with a global batch size of 256. IW-OPD and TA-OPD use the frozen-replay adaptations from the main experiments. The learning rate is 2×10^{-6} with a KL coefficient of 0.01. Evaluation uses temperature 0.6 and top-$p = 0.95$. Table 8 shows that Lightning OPD 2.0 improves the average Pass@1 from 50.0% to 54.1% over Lightning OPD, with gains of 5.6 and 6.0 percentage points on AIME 2025 and HMMT February 2025, respectively. It also exceeds IW-OPD and TA-OPD by 2.9 and 5.1 points in average Pass@1. These results extend the benefits of style-bias correction to more diverse teachers.

Table 8: Additional teacher evaluation with Polaris-4B-Preview and Qwen3-4B-SFT. All entries report Pass@1 in percent. $\diamondsuit$ marks frozen-replay adaptations. Avg. is the mean over the three benchmarks. Boldface marks the highest point estimate in each column.

Method	**AIME 2024**	**AIME 2025**	**HMMT Feb. 2025**	**Avg.**
SFT reference	57.5	52.4	34.9	48.3
Lightning OPD	60.6	55.2	34.2	50.0
IW-OPD$^{\diamondsuit}$	60.4	57.5	35.6	51.2
TA-OPD$^{\diamondsuit}$	56.9	56.5	33.5	49.0
Lightning OPD 2.0	**61.3**	**60.8**	**40.2**	**54.1**

F Limitations

Due to computational constraints, our experiments focus on student models with fewer than 10B parameters and are limited to mathematical reasoning and code generation tasks. We plan to extend Lightning OPD 2.0 to larger student models and a broader range of tasks, including agentic reasoning and tool use.