LAYER AGREEMENT FOR TARGET REFORMULATION IN ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) supervises a student on its own trajectories, but large teacher–student distribution gaps can destabilize direct alignment with the teacher's final predictions. This raises a target-construction question: what distribution should the student learn at its current state? We find that directional agreement between teacher and student probability revisions across layers is strongly associated with smaller final-distribution mismatch, providing evidence of local behavioral compatibility. Motivated by this observation, we propose *Layer Agreement for Target Reformulation* (LaTR), which uses layer evidence to construct an intermediate alignment target. At each student-visited prefix, target construction uses two signals. *Teacher guidance* uses the final teacher–student probability gap to specify each candidate token's correction direction. *Layer evidence* uses intermediate-to-final probability changes to assess both models' support for that direction. Combining these signals yields a bounded adjustment to teacher logits that reshapes both candidate-token selection and relative target probabilities. The student then aligns with a teacher-anchored distribution informed by both models' local behavior to mitigate alignment instability caused by large teacher–student distribution gaps. We evaluate three students across seven mathematical reasoning and code-generation benchmarks. LaTR achieves the best reported overall averages and code scores among compared methods, with gains over the strongest baseline of up to 3.81 percentage points in the seven-benchmark average and 4.70 percentage points on LiveCodeBench-v6.

1 INTRODUCTION

Large language models support open-ended generation, mathematical and code reasoning, and tool use, but their reasoning and deployment costs grow rapidly with model scale and inference length (Yang et al., 2025; Wu et al., 2025; Poddar et al., 2025). Knowledge distillation therefore transfers their capabilities to smaller, more efficient generative models (Jiang et al., 2025; Hinton et al., 2015). Conventional distillation teacher forces the student on teacher-generated trajectories. On-policy distillation (OPD) instead lets the student generate a prefix and queries the teacher on states the student actually visits (Agarwal et al., 2024; Jia et al., 2026; Fu et al., 2026; Li et al., 2026a). This placement of supervision reduces the train–inference mismatch between teacher and student trajectories and provides dense local feedback at states the student will encounter at deployment. Prior work shows that OPD and related generative distillation methods improve generation quality, reasoning ability, and training efficiency (Ko et al., 2024; Xu et al., 2025; Zhang et al., 2026).

However, OPD creates a finer alignment challenge: on a student-visited prefix, the teacher must turn its predictions into local vocabulary-level supervision the student can absorb (Yang et al., 2026b; Xu et al., 2025; Zhang et al., 2026). Differences in capacity and reasoning trajectory can move the student prefix away from the teacher's high-probability region, producing a large distribution gap (Li et al., 2026a; Niu et al., 2025). Tokenizer and representation-space mismatches further complicate cross-model alignment (Chen et al., 2025; Yang et al., 2026a). The teacher's final distribution can then be an unreliable immediate target. Blindly fitting it can weaken distillation (Zhang et al., 2023), while large teacher–student gaps can produce extreme updates for student-sampled tokens in low-probability teacher regions and cause reverse-KL diversity collapse (Ko et al., 2026; Jang et al., 2026). To improve supervision reliability under teacher–student mismatch, existing responses adjust supervision strength through token importance or position reliability (Xu et al., 2026; Liu et al., 2026a), select valuable positions or teacher signals (Wang et al., 2026; Li et al., 2026c), or change the divergence itself (Gu et al., 2024; Jung et al., 2025; Ko et al., 2024). Yet these strategies control where and how strongly the student follows the teacher, rather than reformulating the target

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.

distribution. This leaves a fundamental question unanswered: *is the teacher's final distribution an appropriate target for the student's current state?*

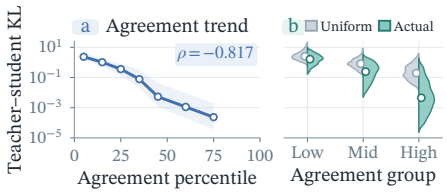

Figure 1: (a) Higher layer agreement is associated with lower teacher–student KL. (b) KL distributions within separately filtered agreement groups. Both keep final predictions fixed, using actual intermediate distributions (*Actual layers*) or uniform distributions over the same candidate tokens (*Uniform reference*).

Recent work suggests that a model's layer-wise predictions carry supervisory signal beyond its final output. OISD incorporates advantage-weighted alignment of intermediate and final-layer predictions into on-policy self-distillation, while OPRD aligns their layer representations on on-policy rollouts (Liu et al., 2026b; Yang et al., 2026a). These signals, however, are exploited within a single model and do not address whether, at a given student state, the teacher's final distribution is what the student should learn immediately. We extend this perspective to teacher–student alignment and ask a more actionable question: instead of only changing how strongly the student fits the teacher's final-layer distribution, can we read both models' layer-wise behavior to judge whether that distribution is locally suitable, and adaptively reshape it into a target better matched to the student's current state? To probe this, we conduct two observational studies on student-visited prefixes, comparing teacher and student predictions over a shared candidate-token set (details in Appendix F). The first study asks whether the teacher and student revise candidate token probabilities in compatible directions and whether this behavior tracks their mismatch. We define *directional agreement* as the fraction of total probability revision on candidate tokens that both models strengthen or both suppress from an intermediate layer to the final layer. It measures how consistently the two models revise their predictions across layers. Figure 1(a) shows that stronger agreement is associated with smaller final-distribution divergence. *This identifies revision agreement as an indicator of local behavioral compatibility.* The second study asks whether this indicator relies on the models' *actual* intermediate predictions or only on their final outputs. Replacing the intermediate predictions with an uninform reference measurably weakens the association (Figure 1(b)). *Actual intermediate predictions therefore contribute information beyond the final outputs.* Together, these findings identify the direction of layer-wise revision as evidence of local teacher–student compatibility, motivating its use in constructing the alignment target.

Building on these observations, we propose **LaTR**, which mitigates alignment failures under large teacher–student gaps by constructing an intermediate target that preserves teacher knowledge while incorporating teacher–student layer agreement. At each student-visited prefix, LaTR combines two signals. *Teacher guidance*, read from the final teacher–student probability gap, sets a correction direction for each candidate token, namely whether the student should increase or decrease its probability. *Layer evidence*, read from each model's probability change between an intermediate layer and the final layer, indicates whether that model's own evolution follows the correction direction and informs the adjustment magnitude. Comparing the two, LaTR reinforces corrections that both models support and downweights conflicting guidance. We implement this rule through a bounded adjustment to the teacher logits. The adjustment reshapes Top-K selection and the relative target probabilities after renormalization. It also bounds the target's KL deviation from the teacher on the selected support. The resulting intermediate target combines teacher knowledge with teacher–student behavioral agreement, learning from which mitigates alignment failures caused by large distribution gaps. Our evaluation covers three Qwen student models spanning 0.6B–7B parameters and seven benchmarks in mathematical reasoning and code generation. LaTR achieves the best overall and code-generation performance among the compared methods, with gains over the strongest baseline of up to 3.81 percentage points in the seven-benchmark average and 4.70 percentage points on LiveCodeBench-v6. Our main contributions are:

- **Rethinking the alignment target.** Under large teacher–student distribution gaps, we extend OPD beyond weighting teacher supervision to explicitly designing the target distribution the student should match.
- **Agreement-guided target reformulation.** We use the teacher–student gap for the correction direction and shared-space layer dynamics as candidate-token-level evidence to construct an intermediate target that preserves teacher information and better fits the student's current state.
- **Cross-task and cross-model improvements.** Evaluated on three Qwen students, LaTR achieves the highest reported seven-benchmark averages and code scores among the compared methods.

2 Related Work

On-Policy and Vocabulary-Level Distillation. On-policy distillation (OPD) trains a student on its own trajectories with teacher feedback, reducing the train–inference mismatch induced by fixed teacher trajectories, as formalized by GKD (Agarwal et al., 2024). MiniLLM emphasizes high-probability teacher modes through reverse KL, while DistiLLM combines skew KL with adaptive off-policy training (Gu et al., 2024; Ko et al., 2024). Recent work studies when and where OPD supervision is useful. Rethinking OPD associates successful transfer with compatible reasoning patterns and novel teacher capabilities (Li et al., 2026a), while prefix-based OPD supervises only student-generated reasoning prefixes (Zhang et al., 2026). TIP (Xu et al., 2026) selects positions using student entropy and teacher–student divergence. Token Teachability further combines local divergence with student-support coverage to retain learnable signals (Wang et al., 2026). These methods select states or positions while retaining the teacher distribution as the target. Vocabulary-level supervision presents a cost–information trade-off. Full-vocabulary matching preserves relations among candidate tokens but is expensive, whereas sampled-token supervision discards them. Revisiting OPD uses Top-K local-support matching (Fu et al., 2026), and Tail-Aware Top-k OPD preserves aggregate probability mass outside the retained support (Huang & Wei, 2026). Related work uses importance sampling to avoid Top-K caching bias or maps distributions across tokenizers (Anshumann et al., 2025; Chen et al., 2025). These studies design supervision positions or candidate-token pools. In contrast, LaTR takes both as input and uses correction directions from final probability differences and layer agreement to construct relative target probabilities, optionally selecting the final support from that pool.

Training Instability in On-Policy Distillation. Large differences between teacher and student output distributions can destabilize OPD. Student-sampled tokens with negligible teacher probability produce extreme log-ratio signals and unreliable updates (Ko et al., 2026). To moderate supervision, EGRSD downweights high-entropy teacher signals in on-policy self-distillation, while Token Teachability selects learnable tokens using KL disagreement and student-support compatibility (Ke et al., 2026; Wang et al., 2026). Beyond weighting and selection, TrOPD uses reverse KL for teacher-compatible predictions and forward KL for outliers (Xing et al., 2026), while Veto constructs an intermediate target in logit space to suppress harmful updates and control diversity loss (Jang et al., 2026). These approaches rely on final-output statistics, leaving the evolution of candidate token preferences across layers unexamined. Complementary work incorporates internal signals into on-policy training. OPRD aligns teacher and student hidden representations on student-generated rollouts (Yang et al., 2026a). OISD adds advantage-weighted alignment of intermediate and final-layer predictions and attention in self-distillation (Liu et al., 2026b). PHF matches a privileged teacher's hidden transition directions and trajectory geometry along the rollout (Li et al., 2026b). In contrast, prior work directly supervises model-internal signals, whereas LaTR jointly evaluates both models' layer revisions against the final gap direction to reshape teacher guidance.

3 Layer Agreement for Target Reformulation

At each student-visited prefix h_t, both models evaluate the same candidate-token pool $\mathcal{P}_t$, a finite set of next-token identities used for local distribution alignment at that position. LaTR uses final teacher–student probability differences to decide which candidate tokens to strengthen or suppress. Both models' intermediate-to-final changes then show whether their layer-wise behavior agrees with these directions (Section 3.1). It then applies bounded teacher-score adjustments, selects the final support, and optimizes a weighted local reverse KL (Section 3.2). Figure 2 summarizes the method, Appendix A lists all notation, and full pseudocode is provided in Appendix T.

3.1 Direction and layer-agreement evidence

This subsection defines how LaTR adjusts the teacher target. Final teacher–student probability differences specify which candidate tokens to strengthen or suppress, and both models' intermediate-to-final changes measure agreement with these directions. Their calibrated and clipped combination yields bounded teacher-logit adjustments.

For a student-visited prefix h_t and model $X \in \{T, S\}$, let H_X^l denote its layer-l hidden state, L_X its final layer, and $m_X < L_X$ the selected intermediate layer. Every vocabulary readout applies the model's final normalization and output head, $z_X^l = \mathrm{LMHead}_X(\mathrm{Norm}_X(H_X^l))$. Here Norm_X denotes the model's final normalization layer. We normalize each readout over the same candidate-

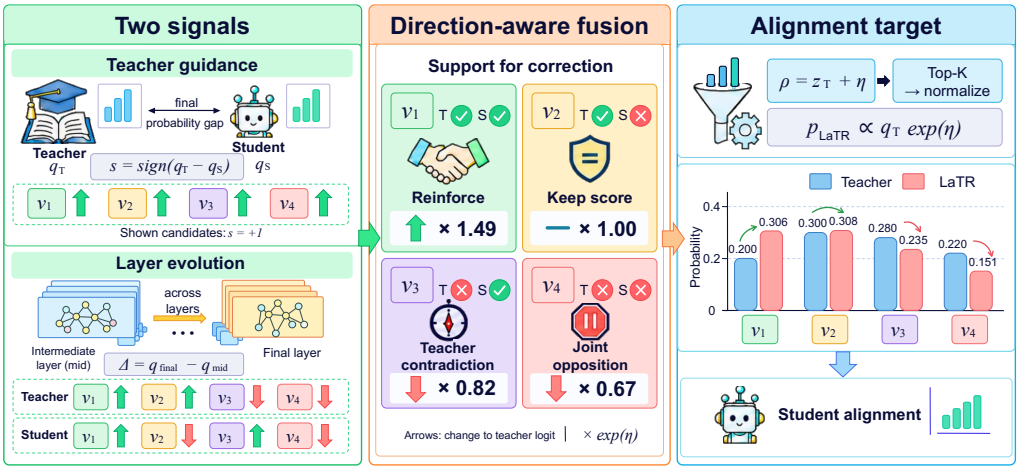

Figure 2: **Overview of LaTR. Left:** Teacher guidance sets correction directions, and layer evolution indicate support. **Middle:** In the illustrated cases, joint support raises scores, teacher-only support preserves them, and teacher contradiction lowers them. **Right:** We apply the adjustments to teacher logits, then jointly normalize the resulting scores over selected candidate tokens into relative target probabilities for student alignment.

token pool $\mathcal{P}_t$, obtaining $\tilde{q}_X^l = \mathrm{softmax}(z_X^l[\mathcal{P}_t])$, and omit conditioning on h_t and $\mathcal{P}_t$ below. For each candidate token v, we define the final teacher–student gap as $g_t(v) := \tilde{q}_T^{L_T}(v) - \tilde{q}_S^{L_S}(v)$. Its sign determines the correction direction:

$$s_t(v) = \mathrm{sign}(g_t(v)). \tag{1}$$

Here $s_t(v) = +1$ asks the student to increase v, while $s_t(v) = -1$ asks it to decrease v, with $\mathrm{sign}(0) = 0$. We set $\delta = 10^{-6}$ and take $s_t(v) = 0$ when $|g_t(v)| \le \delta$, so candidate tokens whose final probabilities are already nearly matched receive no further correction. For each $X \in \{T, S\}$, define the intermediate-to-final change $\Delta_X(v) = \tilde{q}_X^{L_X}(v) - \tilde{q}_X^{m_X}(v)$. Its alignment with the required direction is

$$u_X(v) = s_t(v)\Delta_X(v). \tag{2}$$

Positive $\Delta_X(v)$ means that later layers strengthen v, and $u_X(v) > 0$ means that this evolution agrees with the correction direction. We combine the teacher's and student's alignment with this direction into a reliability score that controls the adjustment magnitude:

$$r_t(v) = \underbrace{\min([\,u_T(v)\,]_+, [\,u_S(v)\,]_+)}_{A_t(v)} - \underbrace{[\,-u_T(v)\,]_+}_{B_t(v)}, \quad [x]_+ = \max(x, 0). \tag{3}$$

The first term, $A_t(v)$, is positive only when both models move v in the correction direction and is limited by the smaller alignment. The second, $B_t(v)$, is positive when the teacher's layer change opposes that direction. Thus, $r_t(v) = A_t(v) - B_t(v)$ combines both models' directional alignment when adjusting the teacher score. For $g_t(v) > 0$, we distinguish four cases:

1. **Teacher and student support** ($u_T(v) > 0, u_S(v) > 0$). Both internal trajectories follow the required direction, giving $A_t(v) > 0$, $B_t(v) = 0$, and $r_t(v) > 0$. The consensus reward increases the teacher score of v, favoring its selection for Top-K support.
2. **Teacher-only support** ($u_T(v) > 0, u_S(v) < 0$). Both terms are zero, so $r_t(v) = 0$. We retain the raw teacher score without an extra bonus or penalty: this may be teacher knowledge the student has not yet acquired, but its compatibility with the student is uncertain. After normalization on the selected support, its probability can rise when other weights decrease in aggregate.
3. **Student-only support** ($u_T(v) < 0, u_S(v) > 0$). The reward is zero and $B_t(v) = |u_T(v)|$, hence $r_t(v) < 0$. The resulting negative adjustment lowers the adjusted teacher logit, which is used for Top-K selection and target probabilities.
4. **Neither model supports** ($u_T(v) < 0, u_S(v) < 0$). Again $A_t(v) = 0$, $B_t(v) = |u_T(v)|$, and $r_t(v) < 0$. Both trajectories oppose the correction, so the negative adjustment reduces the adjusted teacher logit used for selection and target probabilities.

For $g_t(v) < 0$, the score adjustments reverse: positive $r_t(v)$ lowers the teacher score, whereas negative $r_t(v)$ raises it. If either directional signal is zero, $r_t(v) = 0$ unless $u_T(v) < 0$, which retains the teacher-contradiction penalty. Appendix N discusses these cases further.

Raw probability revisions vary in scale across prefixes, layers, and models. We therefore calibrate the reliability score r_t by its mean absolute magnitude within the candidate-token pool and form the bounded log-score adjustment

$$\eta_t(v) = \gamma s_t(v)\bar{r}_t(v), \qquad \bar{r}_t(v) = \mathrm{clip}\left(\frac{r_t(v)}{\frac{1}{|\mathcal{P}_t|}\sum_{u\in\mathcal{P}_t}|r_t(u)| + \epsilon}, -1, 1\right). \tag{4}$$

Pool-wise normalization calibrates $r_t(v)$ before clipping to obtain $\bar{r}_t(v)$, reducing sensitivity to cross-prefix scale differences while preserving the sign and relative strength of layer support when ϵ is small relative to the pool mean. With $s_t(v)$ setting the correction direction and γ the overall scale, $\eta_t(v)$ combines them into a signed teacher-logit adjustment. Since $|\bar{r}_t(v)| \le 1$, each teacher log-score changes by at most γ. Section 3.3 bounds the resulting target deviation, local gradient change, and teacher mass after reselection. We initialize $\gamma = 1$ and analyze its scale and schedule in Section 4.4.

3.2 Teacher-anchored target and local objective

This subsection turns the per-candidate-token adjustment η_t into the actual supervision. Adding it to the teacher logits selects the final Top-K support and defines a teacher-anchored target distribution, which the student then matches under a confidence-weighted local reverse KL.

We apply the bounded correction as $\rho_t(v) = z_T^{L_T}(v) + \eta_t(v)$ for $v \in \mathcal{P}_t$ and select the K candidate tokens used in the loss as $\mathcal{K}_t = \mathrm{TopK}_{v\in\mathcal{P}_t}\,\rho_t(v) \subseteq \mathcal{P}_t$, with $1 \le K \le |\mathcal{P}_t|$. The correction can promote a candidate token outside the teacher's original Top-K into $\mathcal{K}_t$ and simultaneously alter the relative probabilities of the selected candidate tokens. Normalizing the selected scores gives

$$p_{\mathrm{DA}}(v \mid h_t, \mathcal{K}_t) = \mathrm{softmax}(\rho_t[\mathcal{K}_t])_v = \frac{\widetilde{q}_T^{L_T}(v)\exp(\eta_t(v))}{\sum_{u\in\mathcal{K}_t}\widetilde{q}_T^{L_T}(u)\exp(\eta_t(u))}, \qquad v \in \mathcal{K}_t. \tag{5}$$

Since the teacher probability $\widetilde{q}_T^{L_T}(v)$ is proportional to $\exp(z_T^{L_T}(v))$, adding $\eta_t(v)$ to the logit $z_T^{L_T}(v)$ scales its probability weight by $\exp(\eta_t(v))$. Thus, it adjusts the final alignment target according to the teacher's and student's layer-wise support for the required correction.

After reformulating the alignment target, we further weight each token position using teacher confidence and selected-support coverage:

$$w_t = \kappa_t c_t, \quad \kappa_t = 1 - \frac{H(q_T^{L_T})}{\log|\mathcal{V}|}, \quad c_t = \sum_{v\in\mathcal{K}_t} q_T^{L_T}(v). \tag{6}$$

Here $q_T^{L_T}$ is the full-vocabulary teacher distribution and H denotes entropy. The confidence factor κ_t is larger for more concentrated teacher distributions, while c_t is the teacher mass retained by the selected K-token support $\mathcal{K}_t$. Their product assigns greater distillation weight only when the teacher is confident and the selected candidate tokens retain substantial teacher mass. With $\widehat{q}_S^{L_S}$ denoting the student final-layer distribution renormalized on $\mathcal{K}_t$, the loss is

$$\mathcal{L}_t^{LaTR} = w_t D_{\mathrm{KL}}\left(\widehat{q}_S^{L_S}(\cdot \mid h_t, \mathcal{K}_t)\middle\| p_{\mathrm{DA}}(\cdot \mid h_t, \mathcal{K}_t)\right). \tag{7}$$

Through this optimization, the student learns to match an adjusted teacher target that adapts guidance according to the models' layer-wise agreement and joint support for the correction direction. The bounded adjustment retains the teacher as an anchor while aiming to ease alignment across large distribution gaps.

3.3 Analysis of the target construction

LaTR uses bounded logit shifts to adapt teacher guidance while retaining the teacher distribution as an anchor. The following bounds characterize their effects on the target, local gradient, and retained teacher mass (Appendix O).

Since $|\eta_t(v)| \le \gamma$, each teacher log-score changes by at most γ, and the log probability ratio of any two selected candidate tokens changes by at most 2γ. The following proposition bounds the resulting deviation of the target distribution after normalization (Proposition O.1).

Proposition 3.1 (Target-deviation bound). *Let $q_T^{\mathcal{K}_t}$ denote the teacher distribution renormalized on the selected support $\mathcal{K}_t$. Then*

$$D_{\mathrm{KL}}(p_{\mathrm{DA}} \| q_T^{\mathcal{K}_t}) \le \gamma^2/2. \tag{8}$$

For all $u, v \in \mathcal{K}_t$, the target log probability ratio differs from the teacher's by at most 2γ. The quadratic bound is locally tight for bounded exponential reweighting as $\gamma \to 0$.

Two guarantees further characterize the local training signal and support selection. First, reweighting the teacher target on the same support under fixed supervision changes the local reverse-KL gradient with respect to selected student logits by at most $w_t\gamma$ in ℓ_1 norm (Proposition O.2). Second, when the candidate-token pool contains the teacher's full-vocabulary Top-K, the omitted teacher mass after reselection is at most $e^{2\gamma}$ times the original Top-K tail mass (Proposition O.3). Together, these bounds anchor the adjusted target to teacher predictions, limit correction-induced gradient changes, and control teacher-mass loss during reselection, even across large teacher–student distribution gaps.

4 Experiments

4.1 Experimental Setup

Datasets and model pairs. The main evaluation covers three Qwen students spanning 0.6B–7B parameters on 7 mathematical reasoning and code-generation benchmarks. For Qwen2.5-7B-Instruct (Qwen Team, 2025), we use the public OpenThinker3-7B (OpenThoughts, 2025) as the math teacher and our self-trained Qwen2.5-Coder-7B-AgentRL-GRPO-TCR as the code teacher. Both Qwen3-0.6B and Qwen3-1.7B (Yang et al., 2025) use our self-trained Qwen3-GRPO_teacher-4B for both tasks. Appendix B provides model and teacher construction details. Math training uses the English DAPO-Math-17K split (Yu et al., 2025a), with evaluation on AIME24, AIME25, and AMC23 (MAA, 2025), MATH-500 (Hendrycks et al., 2021; Lightman et al., 2023), Minerva (Lewkowycz et al., 2022), and OlympiadBench (He et al., 2024). A fixed subset is held out from training for hyperparameter selection. The final benchmarks are not used for selection. Code training uses the Open-AgentRL split and its fixed Open-AgentRL-30K pool (Yu et al., 2025b). Evaluation reports execution accuracy on LiveCodeBench-v6 (Jain et al., 2024).

Training and evaluation settings. Unless specified otherwise, both tasks use 8 NVIDIA A100 GPUs, a student learning rate of 2×10^{-6}, global batch size 8, 8 responses per prompt, and maximum prompt/response lengths of 2,560/20,480 tokens. For code, all distillation methods first apply supervised fine-tuning (SFT) to both Qwen3 students to learn the required output format. LaTR defaults to $K = 32$, a teacher Top-P candidate-token pool with $P = 2K$, teacher and student readouts at relative depth 0.25, initial correction scale $\gamma = 1$, and $\epsilon = 10^{-8}$. Both tasks linearly decay γ, reducing layer evidence's influence and bringing the target closer to the teacher's final-layer distribution. Evaluation uses temperature 1.0 and top-$p = 0.6$. We report five-run means of math pass@1 and LiveCodeBench-v6 avg@32 using its official execution checker. Additional math avg@32 results appear in Appendix R, and task prompts in Appendix U.

Baselines and variants. (1) **Vanilla** is the undistilled base model; (2) **sampled-token OPD** supervises only the sampled token; (3) **Top-K OPD** matches the teacher final-layer distribution on a sparse Top-K support; (4) **OPSD** uses the same backbone to form teacher and student distributions with and without a privileged reference solution, respectively; and (5) **Veto** constructs an intermediate target from teacher and student scores. Appendix C gives full definitions.

4.2 Main results

Overall performance. Table 1 reports percentages for three models on six math benchmarks and LiveCodeBench-v6, with Avg denoting the unweighted mean over all seven benchmarks. LaTR scores 48.49%, 42.45%, and 52.14% for Qwen2.5-7B, Qwen3-0.6B, and Qwen3-1.7B, respectively. Relative to Top-K OPD, these are gains of 3.81, 2.27, and 2.33 percentage points. On math, LaTR improves over Top-K OPD on five of six benchmarks for Qwen2.5-7B and all six for both Qwen3 students. Its avg@32 results also improve over Vanilla on every benchmark for each Qwen student (Appendix R). On LiveCodeBench-v6, LaTR scores 58.87%, 39.51%, and 39.23%, surpassing Top-K OPD by 4.70, 4.98, and 0.59 points, respectively. *We also provide the results of Llama-3.1-8B-Instruct in Appendix E.* These results support the benefit of target reformulation across model scales and both task domains.

Table 1: Math and code results (%). Avg averages all seven benchmarks. LCB-v6 denotes LiveCodeBench-v6. Entries report mean±std over five evaluation seeds.

Model	Method	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	LCB-v6	Avg
Qwen2.5-7B Instruct	Vanilla	13.24 ± 0.92	9.93 ± 1.72	39.93 ± 0.94	68.23 ± 0.52	28.26 ± 0.76	34.47 ± 0.83	21.08 ± 0.71	30.73
	sampled-token	10.02 ± 0.74	16.77 ± 1.95	44.95 ± 1.02	80.04 ± 0.48	32.33 ± 0.69	43.04 ± 0.81	51.58 ± 0.88	39.82
	OPSD	16.71 ± 1.22	20.06 ± 0.88	49.96 ± 0.97	78.47 ± 0.51	30.76 ± 0.73	39.35 ± 0.85	28.78 ± 0.78	37.73
	Top-K OPD	23.35 ± 2.14	**26.75 ± 1.03**	49.99 ± 0.89	80.45 ± 0.44	34.13 ± 0.64	43.95 ± 0.74	54.17 ± 0.82	44.68
	Veto	20.35 ± 1.31	20.32 ± 1.47	62.53 ± 0.93	78.47 ± 0.56	32.76 ± 0.71	44.53 ± 0.79	51.95 ± 0.86	44.42
	LaTR	**26.73 ± 1.08**	23.06 ± 1.13	**67.51 ± 0.96**	**82.05 ± 0.39**	**34.85 ± 0.77**	**46.36 ± 0.70**	**58.87 ± 0.91**	**48.49**
Qwen3-0.6B	Vanilla	9.36 ± 0.68	14.14 ± 1.12	45.02 ± 1.11	70.78 ± 0.61	21.58 ± 0.82	35.37 ± 0.91	14.89 ± 0.66	30.16
	sampled-token	16.67 ± 1.31	19.96 ± 0.94	50.48 ± 1.06	73.53 ± 0.57	**24.03 ± 0.78**	38.69 ± 0.88	24.65 ± 0.75	35.43
	OPSD	12.58 ± 0.79	17.12 ± 1.46	49.97 ± 1.09	71.31 ± 0.59	23.05 ± 0.80	35.33 ± 0.93	17.73 ± 0.69	32.44
	Top-K OPD	19.36 ± 1.06	25.54 ± 2.18	57.45 ± 0.98	79.22 ± 0.54	21.89 ± 0.74	43.26 ± 0.84	34.53 ± 0.79	40.18
	Veto	16.67 ± 0.88	23.25 ± 1.36	52.54 ± 1.01	72.85 ± 0.63	23.16 ± 0.77	37.55 ± 0.90	36.04 ± 0.81	37.44
	LaTR	**20.31 ± 1.12**	**26.67 ± 1.18**	**62.55 ± 1.12**	**79.66 ± 0.49**	23.43 ± 0.86	**45.04 ± 0.80**	**39.51 ± 0.82**	**42.45**
Qwen3-1.7B	Vanilla	31.36 ± 1.42	19.95 ± 0.89	58.25 ± 0.92	77.07 ± 0.50	30.61 ± 0.71	50.17 ± 0.78	22.73 ± 0.70	41.45
	sampled-token	36.69 ± 2.28	23.84 ± 1.11	70.74 ± 0.86	81.02 ± 0.45	32.12 ± 0.67	37.05 ± 0.83	32.45 ± 0.76	44.84
	OPSD	34.09 ± 0.99	24.75 ± 2.14	68.05 ± 0.90	80.15 ± 0.48	32.16 ± 0.69	39.45 ± 0.81	28.94 ± 0.74	43.94
	Top-K OPD	35.37 ± 1.24	30.66 ± 0.91	72.26 ± 0.82	85.54 ± 0.40	32.29 ± 0.63	53.93 ± 0.72	38.64 ± 0.71	49.81
	Veto	34.68 ± 1.26	31.26 ± 1.04	72.53 ± 0.88	83.62 ± 0.46	34.05 ± 0.75	54.18 ± 0.84	35.02 ± 0.73	49.33
	LaTR	**36.77 ± 1.15**	**33.23 ± 1.19**	**75.51 ± 0.78**	**88.87 ± 0.52**	**34.55 ± 0.79**	**56.82 ± 0.68**	**39.23 ± 0.80**	**52.14**

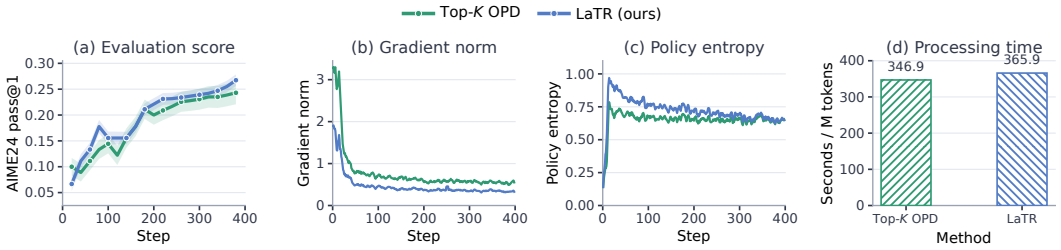

Figure 3: Training dynamics and processing cost.

Target reformulation with token-selection methods. We test whether LaTR target reformulation complements rules that select supervised token positions. We keep each method's selected positions, replace its teacher target with LaTR's layer-guided candidate-token distribution, and omit LaTR position weighting to isolate this change. The six-benchmark average rises by 0.80, 2.93, and 1.20 points over entropy-only, TIP-style, and TA-OPD, respectively (Table 2). Consistent gains across all three rules establish their compatibility with target reformulation. Full results appear in Appendix Q.

Table 2: Compatibility with token-selection methods.

Variant	Avg
Entropy	46.40
Entropy + LaTR target	**47.20**
TIP	44.20
TIP + LaTR target	**47.13**
TA-OPD	46.30
TA-OPD + LaTR target	**47.50**

Training dynamics and cost. In figure 3, LaTR improves AIME24 performance over Top-K OPD. Smaller initial gradient spikes and generally lower gradient norms suggest more stable optimization. Policy entropy is higher early in training, while the gap gradually narrows later. This temporal pattern supports the intended progression of target adaptation: layer evidence reshapes candidate token supervision more strongly early on, while decaying γ reduces this intervention and brings the alignment target closer to the teacher final-layer distribution later. The cost panel compares active processing time per million tokens on Qwen3-1.7B: 365.94 seconds for LaTR versus 346.88 for Top-K OPD, a 5.49% increase alongside the accuracy gains. Appendix S reports additional dynamics results on code, and Appendix P provides the full timing and throughput breakdown.

4.3 Analytical Studies

I. Layer evidence and continuation compatibility. We test whether layer evidence identifies next-token choices that lead to closer teacher–student predictions during subsequent generation. At the same student prefix, we force alternative candidate tokens and let the student generate 20 further tokens for each choice. These continuations reveal how the next-token choice affects the states the student subsequently visits. We measure teacher-to-student KL at every state along each con-

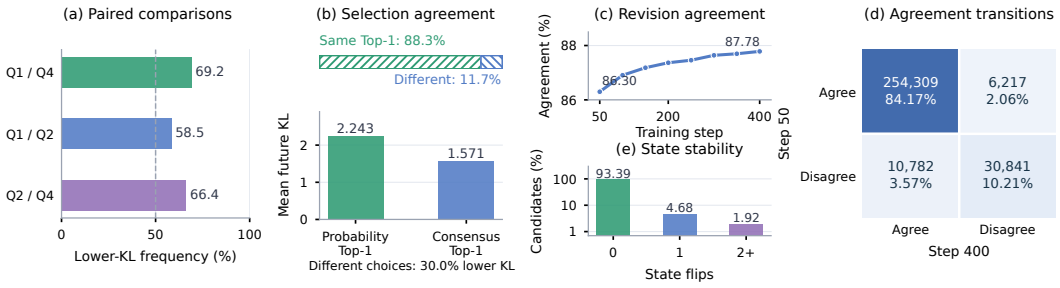

Figure 4: **Forced replacement and agreement dynamics.** (a) Within-prefix pair wins. (b) Top-1 overlap and future KL when selections differ. (c) Revision agreement increases. (d) Conflict resolution exceeds regression. (e) Most candidate tokens remain stable.

tinuation and average it over all 20 states, with lower KL indicating greater compatibility. Each comparison holds the original prefix fixed (Appendix G).

- *Does layer support predict compatible continuations?* At each prefix, Q1–Q4 denote support by both models, the teacher only, the student only, and neither, respectively. We select each group's highest teacher-probability candidate token and compare them within the prefix. Q1 beats Q4 in 69.2% of pairs and Q2 in 58.5%, while Q2 beats Q4 in 66.4% (Figure 4(a)). These within-prefix results show that teacher–student layer support predicts continuation compatibility and provides evidence for target construction.
- *Does layer evidence add information beyond probability?* We compare the probability Top-1 with the layer-consensus Top-1. The two rules select the same candidate token on 88.3% of prefixes. On the remaining 11.7%, forcing the consensus choice reduces mean future KL from 2.243 to 1.571 and wins in 59.3% of pairs (Figure 4(b)). Layer evidence therefore changes few choices, but identifies more compatible alternatives when it does.

II. Evolution and stability of layer agreement. We examine whether teacher–student layer agreement improves during training and whether changes persist across checkpoints. We track fixed teacher Top-32 candidate tokens at checkpoints 50–400, defining agreement as $\Delta_T(v)\Delta_S(v) > 0$. Agreement rises from 86.30% to 87.78% (Figure 4(c)), showing that training gradually aligns the models' layer revisions. Figure 4(d) shows that 3.57% of candidate tokens move from conflict to agreement, compared with 2.06% in the reverse direction. This net shift shows that training resolves more directional conflicts than it creates. Figure 4(e) further shows that 93.39% never switch state, 4.68% switch once, and only 1.92% switch more than once. Thus, established agreement remains stable while the changing subset moves preferentially toward agreement (Appendix H).

III. Teacher–student distribution-shift stress test. We examine how the benefit of target reformulation changes as the teacher distribution moves farther from the fixed student. We take four chronologically ordered checkpoints along the training of our Qwen teacher and train matched Top-K OPD and LaTR students from the same Qwen3-0.6B initialization (Appendix D). The accumulated teacher updates produce progressively different teacher distributions, with teacher accuracy rising from 66.17% to 69.51% and KL increasing from 0.4758 to 0.5035. Top-K OPD rises from 38.07% to 40.24%, whereas LaTR improves from 40.72% to 44.06%. Its relative gain grows from 6.95% to 9.49%, indicating that target reformulation becomes more beneficial as the teacher–student distribution shift increases.

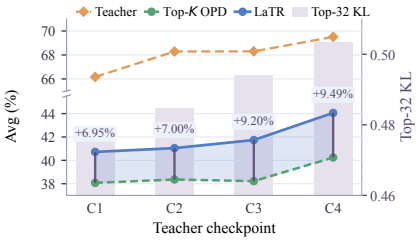

Figure 5: Teacher–student distribution-shift stress test. Bars show the initial teacher-to-student Top-32 KL.

4.4 Ablation Studies

Sensitivity to readout layers. We test how the depth of teacher and student readouts affects target reformulation. We first evaluate all 25 combinations of $m_T, m_S \in \{4, 7, 10, 13, 16\}$ under otherwise fixed settings. Figure 6(a) shows relatively stable accuracy when both layers lie between 4 and 13 (45.73%–46.50%), while moving either readout to layer 16 generally reduces performance. This broad stability shows little dependence on a specific layer pair, so we use the same relative depth for both models and vary it jointly. Figure 6(b) shows the Qwen2.5 and Qwen3 sweeps, while

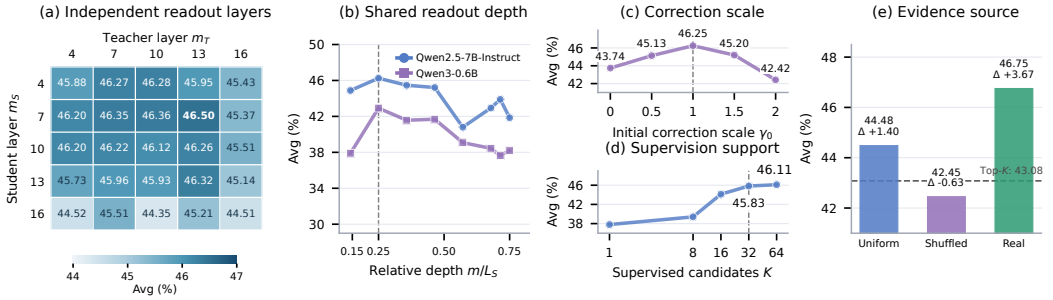

Figure 6: **Ablation study.** (a) Independent teacher and student readouts. (b) Both Qwen models favor an intermediate readout near relative depth 0.25. (c) Performance peaks at $\gamma_0 = 1$. (d) Gains diminish beyond $K = 32$. (e) Actual token-aligned layer evidence performs best.

Appendix I adds the corresponding Llama sweep. Across all settings, the strongest average occurs near one quarter of model depth, and substantially later readouts reduce performance. We therefore set both readouts near relative depth 0.25 by default.

Sensitivity to correction scale. We test how strongly layer evidence should modify the teacher target by varying $\gamma_0 \in \{0, 0.5, 1, 1.5, 2\}$, using linear decay for positive scales. $\gamma_0 = 0$ disables the layer-based adjustment. Figure 6(c) shows that the held-out choice $\gamma_0 = 1$ also achieves the highest six-benchmark average in this post-selection analysis, whereas larger scales of 1.5 and 2 reduce performance. This pattern favors a moderate correction that incorporates layer evidence while remaining close to the teacher's final-layer distribution. Scale results appear in Appendix L.

Number of supervised candidate tokens. We examine how the supervision support size affects performance by varying $K \in \{1, 8, 16, 32, 64\}$ over 400 training steps. Figure 6(d) shows that larger supports improve average accuracy, with diminishing returns beyond $K = 32$. Increasing K from 16 to 32 gains 1.70 percentage points, whereas doubling it again to 64 adds only 0.28 points. This post-selection analysis is consistent with the held-out choice $K = 32$, which approaches the best observed accuracy while supervising half as many candidate tokens as $K = 64$. Appendix M complements this accuracy ablation with a full-vocabulary coverage analysis: unadjusted teacher Top-32 retains 99.72% of probability mass on average on the sampled Qwen2.5 math prefixes.

Source of target-reformulation evidence. We test whether gains require actual intermediate predictions and their correct candidate assignments. Uniform replaces both models' intermediate distributions with uniform distributions over the same candidate pool, retaining their final predictions. Shuffled randomly reassigns measured layer evidence across candidate tokens. Real retains the measured evidence for each token. Uniform gains 1.40 points, compared with 3.67 for Real (Figure 6(e)), indicating that actual intermediate predictions add useful information. Shuffled falls 0.63 points below Top-K OPD, showing that assigning this evidence to the wrong tokens can harm learning. These controls support token-specific layer evidence for target reformulation (Appendix K).

Target reformulation and position weighting. We isolate the gains from changing the teacher target and weighting each token position's loss by enabling each separately and together. Target reformulation alone raises the average from 43.08% to 45.66%, while position weighting alone reaches 43.74%. Combining them reaches 46.25% (Table 3). Thus, the main gain comes from target reformulation, with position weighting adding 0.59 points. Full component results appear in Appendix J.

Table 3: Target reformulation and position weighting.

Target reformulation	Position weighting	Avg
×	×	43.08
×	✓	43.74
✓	×	45.66
✓	✓	**46.25**

5 Conclusion

We introduced LaTR to improve on-policy distillation under large teacher–student distribution gaps. It uses final probability differences to set correction directions and layer-wise probability revisions to adjust teacher logits, producing a bounded teacher-anchored target. Across three Qwen students and seven math and code benchmarks, LaTR achieves the best reported overall averages and code scores among the compared methods. Intervention and ablation results further show that token-specific layer evidence improves target construction, with most gains coming from target reformulation.

AI USE STATEMENT

Generative AI tools assisted with literature search and organization, proof development, code preparation, and language polish. The authors reviewed and validated all AI-assisted outputs, made all final research decisions, and take responsibility for the complete content of this work.

ETHICS STATEMENT

This work uses publicly available language-model checkpoints and benchmark datasets. It does not collect human subjects data or introduce a new human annotation process.

REPRODUCIBILITY STATEMENT

We specify the data, training and evaluation protocols, and hyperparameters. The appendices document teacher construction and baselines, report full ablations and computational costs, provide complete proofs and algorithm pseudocode, and list the task prompts. We use public datasets and checkpoints where applicable.

REFERENCES

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

Anshumann, Mohd Abbas Zaidi, Akhil Kedia, Jinwoo Ahn, Taehwak Kwon, Kangwook Lee, Haejun Lee, and Joohyung Lee. Sparse logit sampling: Accelerating knowledge distillation in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18085–18108. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.885. URL https://aclanthology.org/2025.acl-long.885/.

Yijie Chen, Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. Enhancing cross-tokenizer knowledge distillation with contextual dynamical mapping. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8005–8018. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.419. URL https://aclanthology.org/2025.findings-acl.419/.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. URL https://arxiv.org/abs/2501.12948.

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. *arXiv preprint arXiv:2603.25562*, 2026. doi: 10.48550/arXiv.2603.25562. URL https://arxiv.org/abs/2603.25562.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=5h0qf7IBZZ.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024. doi: 10.18653/v1/2024.acl-long.211. URL https://aclanthology.org/2024.acl-long.211/.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset.

In *Advances in Neural Information Processing Systems*, volume 33, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. URL https://arxiv.org/abs/1503.02531.

Huipeng Huang and Hongxin Wei. Tail-aware top-k on-policy distillation. *arXiv preprint arXiv:2608.14728*, 2026. doi: 10.48550/arXiv.2608.14728. URL https://arxiv.org/abs/2608.14728.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, Kai Dang, Yang Fan, Yichang Zhang, An Yang, Rui Men, Fei Huang, Bo Zheng, Yibo Miao, Shanghaoran Quan, Yunlong Feng, Xingzhang Ren, Xuancheng Ren, Jingren Zhou, and Junyang Lin. Qwen2.5-Coder technical report. *arXiv preprint arXiv:2409.12186*, 2024. URL https://arxiv.org/abs/2409.12186.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024. URL https://arxiv.org/abs/2403.07974.

Ijun Jang, Jewon Yeom, Juan Yeo, Hyunggyu Lim, and Taesup Kim. Stable on-policy distillation through adaptive target reformulation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 42217–42227. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.2094. URL https://aclanthology.org/2026.findings-acl.2094/. Veto; arXiv:2601.07155.

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. *arXiv preprint arXiv:2605.06387*, 2026. URL https://arxiv.org/abs/2605.06387.

Wangyi Jiang, Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. Teach small models to reason by curriculum distillation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7412–7422. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.376. URL https://aclanthology.org/2025.emnlp-main.376/.

Seongryong Jung, Suwan Yoon, DongGeon Kim, and Hwanhee Lee. Todi: Token-wise distillation via fine-grained divergence control. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8078–8091. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.409. URL https://aclanthology.org/2025.emnlp-main.409/.

Junlong Ke, Zichen Wen, Weijia Li, Conghui He, and Linfeng Zhang. Respecting self-uncertainty in on-policy self-distillation for efficient LLM reasoning. *arXiv preprint arXiv:2605.13255*, 2026. doi: 10.48550/arXiv.2605.13255. URL https://arxiv.org/abs/2605.13255.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024. URL https://arxiv.org/abs/2402.03898.

Jongwoo Ko, Sara Abdali, Young Jin Kim, Tianyi Chen, and Pashmina Cameron. Scaling reasoning efficiently via relaxed on-policy distillation. *arXiv preprint arXiv:2603.11137*, 2026. URL https://arxiv.org/abs/2603.11137.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. *arXiv preprint arXiv:2206.14858*, 2022. URL https://arxiv.org/abs/2206.14858.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026a. doi: 10.48550/arXiv.2604.13016. URL https://arxiv.org/abs/2604.13016.

Yuhan Li, Mingxu Zhang, Dazhong Shen, and Ying Sun. PHF: Privileged hidden flow for on-policy self-distillation. *arXiv preprint arXiv:2606.29340*, 2026b. URL https://arxiv.org/abs/2606.29340.

Yuying Li, Leqi Zheng, Yongzi Yu, Wenrui Zhou, Xuchang Zhong, Xing Hu, Jing Jin, Huangjie Yuan, and Tao Feng. Filter, then reweight: Rethinking optimization granularity in on-policy distillation. *arXiv preprint arXiv:2606.02684*, 2026c. URL https://arxiv.org/abs/2606.02684.

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

Xiaogeng Liu, Xinyan Wang, Yingzi Ma, Yechao Zhang, and Chaowei Xiao. When are teacher tokens reliable? position-weighted on-policy self-distillation for reasoning. *arXiv preprint arXiv:2605.21606*, 2026a. doi: 10.48550/arXiv.2605.21606. URL https://arxiv.org/abs/2605.21606. PW-OPSD.

Xinyu Liu, Darryl Cherian Jacob, Yang Zhou, Jindong Wang, and Pan He. OISD: On-policy internal self-distillation of language models. *arXiv preprint arXiv:2605.29089*, 2026b. URL https://arxiv.org/abs/2605.29089.

MAA. American mathematics competitions: AMC 2023 and AIME 2024–2025. Mathematical Association of America, 2025. URL https://maa.org/student-programs/amc/.

Meta. Llama-3.1-8B-Instruct. Hugging Face model card, 2024. URL https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct.

Lujie Niu, Haochen Sun, Fangkun Zhao, Sheng Chen, Zimeng Bai, Jiawei Zhang, Caixia Yuan, and Xiaojie Wang. CoTD-PO: Chain-of-thought distillation with preference optimization. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 19975–19986. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.1087. URL https://aclanthology.org/2025.findings-emnlp.1087/.

OpenThoughts. OpenThinker3-7B. Hugging Face model card, 2025. URL https://huggingface.co/open-thoughts/OpenThinker3-7B.

Soham Poddar, Paramita Koley, Janardan Misra, Niloy Ganguly, and Saptarshi Ghosh. Towards sustainable NLP: Insights from benchmarking inference energy in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 12688–12704. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.632. URL https://aclanthology.org/2025.naacl-long.632/.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025. URL https://arxiv.org/abs/2412.15115.

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. *arXiv preprint arXiv:2605.26844*, 2026. doi: 10.48550/arXiv.2605.26844. URL https://arxiv.org/abs/2605.26844. Teachability-Aware OPD (TA-OPD).

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for llm problem-solving. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8c3caae2f725c8e2a55ecd600563d172-Abstract-Conference.html.

Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation. *arXiv preprint arXiv:2606.01249*, 2026. doi: 10.48550/arXiv.2606.01249. URL https://arxiv.org/abs/2606.01249.

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 *International Conference on Learning Representations*, 2025. doi: 10.48550/arXiv.2410.11325. URL https://arxiv.org/abs/2410.11325.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. TIP: Token importance in on-policy distillation. *arXiv preprint arXiv:2604.14084*, 2026. doi: 10.48550/arXiv.2604.14084. URL https://arxiv.org/abs/2604.14084.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. doi: 10.48550/arXiv.2505.09388. URL https://arxiv.org/abs/2505.09388.

Shenzhi Yang, Guangcheng Zhu, Bowen Song, Haobo Wang, Mingxuan Xia, Xing Zheng, Yingfan Ma, Zhongqi Chen, Weiqiang Wang, Junbo Zhao, and Gang Chen. OPRD: On-policy representation distillation. *arXiv preprint arXiv:2606.06021*, 2026a. doi: 10.48550/arXiv.2606.06021. URL https://arxiv.org/abs/2606.06021.

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan 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, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025a. URL https://arxiv.org/abs/2503.14476.

Zhaochen Yu, Ling Yang, Jiaru Zou, Shuicheng Yan, and Mengdi Wang. Demystifying reinforcement learning in agentic reasoning. *arXiv preprint arXiv:2510.11701*, 2025b. URL https://arxiv.org/abs/2510.11701.

Dongxu Zhang, Zhichao Yang, Sepehr Janghorbani, Jun Han, Andrew II Ressler, Qian Qian, Gregory D. Lyng, Sanjit Singh Batra, and Robert E. Tillman. Fast and effective on-policy distillation from reasoning prefixes. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25553–25569. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1276. URL https://aclanthology.org/2026.findings-acl.1276/.

Rongzhi Zhang, Jiaming Shen, Tianqi Liu, Jialu Liu, Michael Bendersky, Marc Najork, and Chao Zhang. Do not blindly imitate the teacher: Using perturbed loss for knowledge distillation. *arXiv preprint arXiv:2305.05010*, 2023. URL https://arxiv.org/abs/2305.05010. The paper is commonly referred to as PTLoss.

A NOTATION

Table 4 summarizes the notation used in the method and its analysis.

Table 4: Notation used in Section 3.

Symbol	Meaning
Models and local readouts	
h_t	Student-generated prefix before predicting token position t.
$X \in \{T, S\},\ m_X,\ L_X$	Teacher or student model, with its selected intermediate and final layers.
z_X^l	Vocabulary logits read from model X at layer l.
$\mathcal{P}_t,\ P$	Shared candidate-token pool at position t and its size $P = \lvert\mathcal{P}_t\rvert$.
$\widetilde{q}_X^l$	Layer-l distribution of model X, normalized over $\mathcal{P}_t$.
Correction direction and layer evidence	
$g_t(v),\ s_t(v),\ \delta$	Final probability gap, its correction direction, and the dead-zone threshold.
$\Delta_X(v)$	Intermediate-to-final probability change of model X for v.
$u_X(v)$	Direction-aligned layer revision $s_t(v)\Delta_X(v)$.
$A_t(v),\ B_t(v)$	Joint directional support and teacher opposition.
$r_t(v),\ \bar{r}_t(v),\ \epsilon$	Raw reliability, its pool-calibrated clipped value, and the calibration stabilizer.
$\gamma,\ \eta_t(v)$	Adjustment budget and resulting bounded teacher-logit adjustment.
Target construction and optimization	
$\rho_t(v)$	Adjusted teacher logit $z_T^{L_T}(v) + \eta_t(v)$.
$K,\ \mathcal{K}_t$	Number of supervised candidate tokens and the Top-K support selected from $\mathcal{P}_t$ using ρ_t.
p_{DA}	Teacher-anchored target obtained by normalizing adjusted logits on $\mathcal{K}_t$.
$\widehat{q}_S^{L_S}$	Student final-layer distribution renormalized on $\mathcal{K}_t$.
$\kappa_t,\ c_t,\ w_t$	Teacher confidence, retained full-vocabulary probability mass, and their position weight $w_t = \kappa_t c_t$.
$q_T^{\mathcal{K}_t}$	Teacher distribution renormalized on $\mathcal{K}_t$, used in the analysis.

B TEACHER MODELS AND CONSTRUCTION DETAILS

OpenThinker3-7B. For the Qwen2.5 mathematical-reasoning setting, we use the publicly released OpenThinker3-7B checkpoint[1] from Hugging Face (OpenThoughts, 2025). It is a Qwen2.5-7B-Instruct-based[2] reasoning model trained by its authors on OpenThoughts3-1.2M. We use it only as a fixed teacher and do not further update its parameters.

DeepSeek-R1-Distill-Llama-8B. For mathematical reasoning, we initialize the student from Llama-3.1-8B-Instruct[3] (Meta, 2024) and use the public DeepSeek-R1-Distill-Llama-8B[4] (DeepSeek-AI, 2025) as the frozen teacher.

Qwen3-GRPO_teacher-4B. For Qwen3-0.6B[5] and Qwen3-1.7B[6], we train a 4B teacher from Qwen3-4B[7] (Yang et al., 2025) in two stages. First, we perform cold-start supervised fine-tuning on Open-AgentRL-SFT-3K (Yu et al., 2025b), which contains tool-integrated agent trajectories. We then continue from this checkpoint with GRPO on Open-AgentRL-30K, a mixed agentic corpus covering mathematical reasoning, executable code, and tool-use tasks. GRPO uses group-relative rewards from sampled rollouts: each prompt produces 16 responses, groups with no reward variation are excluded, and tool calls are executed in the sandboxed environment used for reward computation.

[1] https://huggingface.co/open-thoughts/OpenThinker3-7B
[2] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct
[3] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
[4] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B
[5] https://huggingface.co/Qwen/Qwen3-0.6B
[6] https://huggingface.co/Qwen/Qwen3-1.7B
[7] https://huggingface.co/Qwen/Qwen3-4B

We use a global batch size of 64, learning rate 10^{-6}, at most 16 interaction turns, and maximum prompt and response lengths of 2,560 and 20,480 tokens. The GRPO stage does not add a reference-KL reward or loss. We merge the resulting distributed checkpoint into a Hugging Face model and denote the frozen teacher used throughout the Qwen3 experiments as QWEN3-GRPO_TEACHER-4B. We will release this teacher checkpoint upon acceptance.

Qwen2.5-Coder-7B-AgentRL-GRPO-TCR. For the Qwen2.5 code setting, we construct a separate teacher from Qwen2.5-Coder-7B-Instruct[8] (Hui et al., 2024). It receives the same cold-start SFT on Open-AgentRL-SFT-3K, followed by tool-integrated GRPO-TCR on Open-AgentRL-30K. This stage uses the same rollout budget, context limits, group-relative reward construction, and sandboxed code execution described above. We merge its actor weights into Hugging Face format and keep the resulting teacher fixed during code distillation. We will release this teacher checkpoint upon acceptance.

C BASELINE DETAILS

Vanilla. We evaluate the undistilled student checkpoint without OPD training.

Sampled-token OPD. At each student-generated position, sampled-token OPD estimates the reverse-KL objective from the token sampled by the student. It provides one teacher-supervised token per position rather than matching a multi-token distribution.

Top-K OPD. Top-K OPD selects the teacher's K highest-probability candidate tokens and renormalizes both models on this shared support. It minimizes the local reverse KL from the student distribution to the teacher distribution without layer-based target adjustment.

OPSD. OPSD performs self-distillation with a shared backbone. A frozen copy conditioned on the problem, reference solution, and student-generated prefix serves as the teacher, while the trainable student receives only the problem and prefix. Their token distributions are aligned along the student rollout.

Veto. Veto (Jang et al., 2026) constructs an intermediate target from the final teacher and student logits. We implement its target as

$$q_{\mathrm{Veto}}(v) \propto \exp(z_T(v) + \beta\ \mathrm{sg}(z_S(v))) \propto q_T(v) q_S(v)^{\beta},$$

where sg stops gradients through the student-logit branch. The product target emphasizes candidate tokens favored by both models and downweights teacher-preferred tokens to which the student assigns little probability. For comparison under the same sparse-computation budget, we apply this target on the teacher Top-K support with $K = 32$. We linearly decay β from 0.8 to 0, returning the target to the teacher distribution. Unlike LaTR, Veto uses final-layer logits only and retains the initial teacher support without layer-guided reselection.

D TEACHER–STUDENT DISTRIBUTION-SHIFT STRESS TEST

We study how the gain from target reformulation changes with teacher–student distribution shift. We use four teacher checkpoints, denoted C1–C4, while fixing the student initialization, architecture, tokenizer, and distillation recipe.

Experimental setup. We select four checkpoints from our trained 4B model, QWEN3-GRPO_TEACHER-4B. Each checkpoint is frozen and paired with the same Qwen3-0.6B initialization. We evaluate every teacher on the six math benchmarks. For each checkpoint, we train matched Top-K OPD and LaTR students for 400 updates with identical data, rollout budgets, optimization settings, and evaluation procedures. LaTR uses $P = 128$, $K = 32$, relative readout depth 0.25, $\gamma_0 = 1$, and linear correction decay. We report all benchmark scores, their mean, and LaTR's relative growth over Top-K OPD.

[8] https://huggingface.co/Qwen/Qwen2.5-Coder-7B-Instruct

Measuring the initial distribution gap. We generate a fixed on-policy prefix bank with the initial Qwen3-0.6B student, comprising 64 questions and five positions per response (320 prefix positions). All four teachers are evaluated on these identical prefixes. At prefix h, let $\mathcal{S}_T^{32}(h)$ be the teacher's final-layer Top-32 support, and renormalize both endpoint distributions on this support:

$$\widetilde{p}_X(v \mid h) = \frac{p_X(v \mid h)}{\sum_{u \in \mathcal{S}_T^{32}(h)} p_X(u \mid h)}, \qquad X \in \{T, S_0\}.$$

We measure the local initial gap as

$$D_{\mathrm{KL}}^{32}(T \| S_0) = \frac{1}{320} \sum_h \sum_{v \in \mathcal{S}_T^{32}(h)} \widetilde{p}_T(v \mid h) \log \frac{\widetilde{p}_T(v \mid h)}{\widetilde{p}_{S_0}(v \mid h)}.$$

This yields 0.4758, 0.4848, 0.4941, and 0.5035 for C1–C4. The support retains more than 99.6% of teacher probability mass and approximately 99.0% of student mass, making the local KL directly representative of the distributions optimized by Top-K OPD.

Table 5: **Teacher–student distribution-shift stress test.** Scores are pass@1 percentages. Growth is the relative improvement of LaTR over Top-K OPD at the same checkpoint.

Condition	Method	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg	Growth
C1	Teacher	56.67	50.00	92.50	94.40	37.50	65.93	66.17	–
	Top-K OPD	13.33	23.33	45.00	78.60	26.10	42.07	38.07	–
	LaTR	16.67	20.00	62.50	80.80	20.96	43.41	40.72	+6.95%
C2	Teacher	63.33	53.33	92.50	94.40	39.34	66.82	68.29	–
	Top-K OPD	13.33	20.00	52.50	80.00	22.31	42.02	38.36	–
	LaTR	13.33	23.30	60.00	80.20	25.00	44.44	41.05	+7.00%
C3	Teacher	56.67	56.67	95.00	93.40	41.54	66.52	68.30	–
	Top-K OPD	10.00	20.00	60.00	77.20	23.16	38.96	38.22	–
	LaTR	13.33	23.33	65.00	81.60	23.90	43.26	41.74	+9.20%
C4	Teacher	63.33	60.00	92.50	94.60	39.34	67.26	69.51	–
	Top-K OPD	20.00	20.00	57.50	78.80	23.90	41.26	40.24	–
	LaTR	23.33	26.67	62.50	82.40	24.43	45.04	44.06	+9.49%

Results and interpretation. From C1 to C4, the measured Top-32 KL increases from 0.4758 to 0.5035. LaTR consistently outperforms its matched Top-K OPD student, and its relative gain increases from 6.95% to 9.49%. Because each pair uses the same student initialization, fixed-prefix gap measurement, and training setup, the widening gain is consistent with greater benefit under a larger local teacher–student distribution gap.

Conclusion. Across C1–C4, larger measured local gaps coincide with larger relative gains from LaTR, supporting layer-guided target reformulation under teacher–student distribution shift.

E LLAMA-3.1-8B-INSTRUCT RESULTS

Table 6 reports the Llama cross-family setting on the six mathematical benchmarks and LiveCodeBench-v6.

F DETAILS OF THE REVISION-AGREEMENT STUDY

Experimental setup. We use Qwen2.5-7B-Instruct as student and OpenThinker3-7B as teacher. The 500 problems come from the math benchmarks in Section 4.1: 9 AIME24, 12 AIME25, 12 AMC23, 152 MATH-500, 88 Minerva, and 227 OlympiadBench problems. We sample 4 responses per problem with temperature 0.7, top-$p = 0.9$, and a 1,024-token limit. For a response of length N, we retain prediction positions $\mathrm{round}(j(N-1)/11)$, $j = 1, \ldots, 11$, giving 11 non-initial prefixes per response and 22,000 prefixes overall. A question-level split assigns 350 problems to calibration and 150 problems (6,600 prefixes) to evaluation. At each prefix h_t, $\mathcal{P}_t$ contains the teacher's final-layer

Table 6: Llama-3.1-8B-Instruct results (%). Avg is the unweighted mean across the six math benchmarks and LiveCodeBench-v6.

Method	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	LCB-v6	Avg
Vanilla	3.33	2.13	22.50	36.00	12.50	14.07	1.04	13.08
sampled-token	16.67	**16.67**	45.00	66.40	23.10	34.37	21.54	31.96
OPSD	10.00	10.00	45.00	56.50	17.31	29.02	16.33	26.31
Top-K OPD	16.67	10.00	50.00	58.00	20.28	30.11	23.18	29.75
LaTR	**16.67**	13.33	**57.50**	**68.20**	**23.90**	**35.85**	**25.63**	**34.44**

Top-32 candidate tokens. Both models' layer-7 and final predictions are independently normalized on this same set:

$$\widetilde{q}_X^l(v \mid h_t, \mathcal{P}_t) = \frac{q_X^l(v \mid h_t)}{\sum_{u \in \mathcal{P}_t} q_X^l(u \mid h_t)}, \qquad X \in \{T, S\}, \quad v \in \mathcal{P}_t. \tag{9}$$

Here q_X^l is the unit-temperature softmax of model X's layer-l logits. Intermediate logits use its final normalization and output head. Final-distribution mismatch is the teacher-to-student $D_{\mathrm{KL}}(\widetilde{q}_T^{L_T} \| \widetilde{q}_S^{L_S})$, in nats, where L_T and L_S denote the final layers. Each observation is one prefix, with agreement and KL computed over its same 32 candidate tokens. The analysis uses recorded prefixes without token replacement or model updates.

Individual revision rates and revision agreement rate. We define $\Delta_X(v) = \widetilde{q}_X^{L_X}(v) - \widetilde{q}_X^{7}(v)$: a positive value means that later layers strengthen candidate token v, and a negative value means they suppress it. The individual revision rate is $M_X = \frac{1}{2}\sum_{v \in \mathcal{P}_t} |\Delta_X(v)|$, the fraction of probability mass redistributed between the two normalized predictions. To measure whether the models revise the same candidate tokens in the same direction, we use

$$A_\Delta(h_t) = \frac{\sum_{v \in \mathcal{P}_t} \omega_t(v) \mathbf{1}[\Delta_T(v)\Delta_S(v) > 0]}{\sum_{v \in \mathcal{P}_t} \omega_t(v)}, \qquad \omega_t(v) = |\Delta_T(v)| + |\Delta_S(v)|. \tag{10}$$

Thus A_Δ is the share of total probability revision assigned to candidate tokens that both models strengthen or both suppress. The sign test counts revisions with the same nonzero direction as agreement.

Question 1: Can layer-wise agreement characterize teacher–student distribution mismatch? Figure 1(a) groups the 6,600 evaluation prefixes using calibration percentiles 0–10, 10–20, 20–30, 30–40, 40–50, and 50–100%. The upper half is pooled because its calibration thresholds equal one. The horizontal coordinate is the percentile-range midpoint, yielding 5, 15, 25, 35, 45, and 75; the upper-half group is plotted at 75. Each point reports median final-layer KL. Table 7 gives the thresholds, counts, and quartiles. The decreasing medians and negative prefix-level Spearman correlation (Table 8) support agreement as an indicator of local distribution compatibility.

Table 7: Groups underlying Figure 1(a). Final-layer KL is in nats.

Calibration percentiles (%)	x	A_Δ upper bound	Prefixes	Median KL	KL interquartile range
0–10	5	0.80463264	598	2.3656	[1.4858, 3.9250]
10–20	15	0.94118421	636	1.0257	[0.57514, 1.8049]
20–30	25	0.98749721	656	0.35983	[0.13111, 0.77588]
30–40	35	0.99863870	662	0.077642	[0.020623, 0.33975]
40–50	45	0.99993763	647	0.0053458	[0.0011682, 0.044623]
50–100	75	1	3,401	2.3539×10^{-4}	[3.6187×10^{-5}, 0.0019656]

Intervals are right-closed, with each preceding upper bound serving as the next lower bound. The first interval includes zero. Displayed bounds are rounded, while assignments use the unrounded calibration thresholds.

Directional agreement or individual revision amount? On the same evaluation prefixes, we compare A_Δ with the individual revision rates M_T and M_S. Each signal is grouped by its own calibration quantiles using the percentile ranges above (Figure 7). Table 8 shows a stronger association

for agreement than for either individual revision rate, supporting the relevance of how the models' revision directions relate to one another.

Question 2: Does this characterization benefit from actual intermediate predictions? We hold the prefixes, candidate-token sets, and final distributions fixed and replace both intermediate readouts with $1/32$. The reference revisions are $\Delta_X^{\mathrm{unif}}(v) = \widetilde{q}_X^{L_X}(v) - 1/32$. Substituting them into the sign test and weights in Eq. 10 gives A_Δ^{unif}. We compare correlation strengths through $G = |\rho(A_\Delta, \mathrm{KL})| - |\rho(A_\Delta^{\mathrm{unif}}, \mathrm{KL})|$ on the full evaluation set. We use 10,000 paired question-cluster bootstrap resamples, retaining all responses and prefixes from each sampled question. Actual-layer agreement reaches $\rho = -0.8167$, compared with -0.7359 for the uniform reference, yielding $G = 0.0808$ (95% CI $[0.0713, 0.0903]$), or an 11.0% relative increase in correlation strength. Actual intermediate readouts therefore provide additional information beyond the shared final predictions.

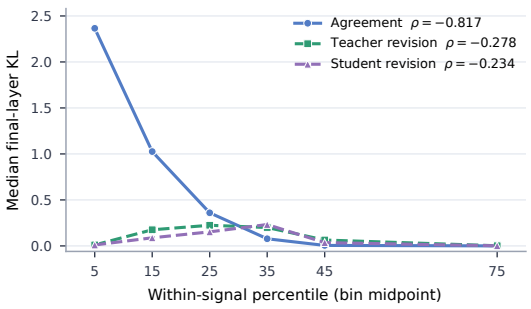

Figure 7: Revision agreement is more strongly associated with teacher–student KL than either model's individual revision rate.

Table 8: Prefix-level correlations and paired gain on the 6,600-prefix evaluation set.

Statistic	Estimate	95% bootstrap interval
Actual-layer agreement: $\rho(A_\Delta, \mathrm{KL})$	-0.8167	$[-0.8271, -0.8055]$
Uniform reference: $\rho(A_\Delta^{\mathrm{unif}}, \mathrm{KL})$	-0.7359	$[-0.7478, -0.7240]$
Teacher revision rate: $\rho(M_T, \mathrm{KL})$	-0.2784	–
Student revision rate: $\rho(M_S, \mathrm{KL})$	-0.2337	–
Paired correlation-strength gain G	0.0808	$[0.0713, 0.0903]$
Relative correlation-strength gain	11.0%	$[9.22\%, 12.80\%]$

Distributions in Figure 1(b). Panel (b) compares uniform-reference and actual-layer KL distributions across lower, middle, and upper agreement terciles. For each signal, prefixes with agreement below one are grouped into terciles after applying a 1.5-IQR filter to $\log_{10}(\mathrm{KL})$. The actual-layer half violins show the central 90% of each tercile, while the uniform-reference half violins use the full filtered terciles. Dots and bars mark medians and interquartile ranges. Table 9 reports the displayed group summaries. Both signals show lower KL in higher-agreement terciles.

Table 9: Displayed nonsaturated groups in Figure 1(b). KL is in nats.

Setting	Tercile	Median KL	KL Q_1	KL Q_3
Uniform reference $\rho = -0.8335$	Lower	2.5052	1.6902	3.9344
	Middle	0.81640	0.49156	1.4058
	Upper	0.19568	0.089060	0.48928
Actual layers $\rho = -0.8714$	Lower	1.6311	0.87399	2.7727
	Middle	0.24294	0.073286	0.59136
	Upper	0.0044939	0.00088695	0.028138

Implication for target construction. Together, the two comparisons motivate using layer-wise directional agreement as evidence of local teacher–student compatibility. The uniform-reference comparison shows that actual intermediate readouts strengthen this association beyond the final-distribution inputs shared by both measures. The final probability gap specifies the desired correction, while each model's intermediate-to-final revisions indicate whether its internal evolution supports that direction. LaTR combines these signals to construct an intermediate alignment target that retains teacher guidance while incorporating teacher–student behavioral agreement.

G Full Results of the Forced-Replacement Study

Setup and candidate-token construction. We use Qwen2.5-7B-Instruct as the student and OpenThinker3-7B as the frozen teacher. From 500 math questions, we sample four student rollouts per question and retain approximately 24,000 student-visited prefixes. At each prefix, both models are evaluated on the same pool of the teacher's final-layer Top-128 tokens, with intermediate readouts $m_T = m_S = 7$. We compute the final-gap direction $s_t(v)$ and the teacher and student directional supports $u_T(v)$ and $u_S(v)$ on the original prefix, before any counterfactual replacement. Candidate tokens are then assigned to Q1 ($u_T > 0, u_S > 0$), Q2 ($u_T > 0, u_S < 0$), Q3 ($u_T < 0, u_S > 0$), or Q4 ($u_T < 0, u_S < 0$). From each nonempty quadrant, we retain the candidate token with the highest teacher final-layer probability, yielding at most four candidate tokens for a prefix. The analysis uses the positive-correction subset $s_t(v) = +1$.

Counterfactual continuation and future KL. For every retained candidate token v at prefix h_t, we replace the next token with v, obtaining $\tilde{h}_{t+1} = [h_t, v]$, and let the student generate a 20-token continuation. At each of the $H = 20$ post-replacement prediction states, the teacher and student are queried on the identical counterfactual context. We define the outcome as

$$\mathrm{FKL}_t(v) = \frac{1}{H} \sum_{j=1}^{H} D_{\mathrm{KL}}\Big(q_T(\cdot \mid \tilde{h}_{t+j}) \Big\| q_S(\cdot \mid \tilde{h}_{t+j})\Big). \tag{11}$$

Thus, lower future KL means that the two models' next-token distributions are closer along the continuation induced by the forced candidate token. It is a local continuation-compatibility measure, not an answer-accuracy measure. This procedure yields 54,650 candidate-token-level future-KL annotations from 771,734 original candidate token records.

Within-prefix comparisons and statistics. For a quadrant comparison A/B, we retain only original prefixes containing an eligible candidate token in both A and B. We force the two candidate tokens separately from that same h_t and compare $\mathrm{FKL}_t(v_A)$ with $\mathrm{FKL}_t(v_B)$. The probability–consensus comparison follows the same rule: on each prefix, each ranking selects one candidate token from the common pool, and we retain a pair only when the selections differ. Each paired difference therefore holds fixed the question, rollout, original prefix, and token position. Only the forced candidate token changes. No result compares A and B across different prefixes. Because a prefix need not contain every quadrant, the matched prefix set can differ across Q1/Q4, Q1/Q2, and Q2/Q4. We report the pair count, paired means, mean paired difference, and the fraction of pairs with lower KL for the first candidate token.

Layer support and subsequent agreement. Table 10 provides the full quadrant comparisons. Q1 improves on Q4 by 0.168 mean KL and on Q2 by 0.073. Q2 improves on Q4 by 1.219 on its corresponding paired subset. The Q1–Q2 comparison supports the value of student compatibility alongside teacher support, while Q2–Q4 shows that teacher-only support remains informative. Together, these comparisons motivate rewarding jointly supported corrections while retaining teacher-supported corrections that lack current student support.

Table 10: Prefix-paired future-KL comparisons ($s_t = +1$). A/B follow the comparison order. A lower denotes the fraction of pairs favoring A.

Comparison	n	KL A	KL B	A–B	A lower
Q1 vs Q4	1762	0.510	0.678	-0.168	69.2%
Q1 vs Q2	1565	0.838	0.910	-0.073	58.5%
Q2 vs Q4	1912	2.397	3.616	-1.219	66.4%

Information beyond student probability. We compare selection by student final-layer probability with selection by the layer-consensus score within each prefix. The two rules select the same candidate token on 4,133 of 4,679 prefixes (88.3%) and different tokens on 546 (11.7%). On the latter subset, consensus reduces mean future KL by 0.672, or 30.0% (Table 11), with a 59.3% win rate. This comparison tests the signal precisely where it changes the decision made by probability alone.

Table 11: Mean future KL on the 546 disagreement prefixes.

Selection rule	Future KL
Student probability	2.243
Layer consensus	**1.571**

H Full Results of Layer Agreement across Training

Question and fixed-prefix protocol. We examine whether student probability revisions increasingly agree with the teacher during training, whether initially conflicting candidate tokens become aligned, and whether agreement states remain stable across checkpoints. We use Qwen2.5-7B-

Instruct and OpenThinker3-7B, fixing 500 step-400 evaluation rollouts and 19 positions per response (5%–95% in 5% increments). The resulting 9,500 prefixes and teacher-final Top-32 supports define 304,000 fixed (prefix, position, token) instances evaluated at steps 50, 100, 150, 200, 250, 300, 350, and 400. Layer-7 and final readouts are separately normalized on this support. Agreement is $\Delta_T(v)\Delta_S(v) > 0$, with revisions active when their magnitudes exceed 10^{-7}.

Matched candidate-token sets. Each comparison follows the same candidate tokens at the relevant checkpoints, so changes in agreement are not caused by a changing active set. Trend and flip analyses use tokens active in the teacher and at all eight student checkpoints, retaining 301,510 instances (99.18%). First-to-last transitions use the corresponding step-50/400 matched sets. A flip is a change of agreement state between adjacent checkpoints. We compute 95% intervals with 2,000 question-cluster bootstrap resamples.

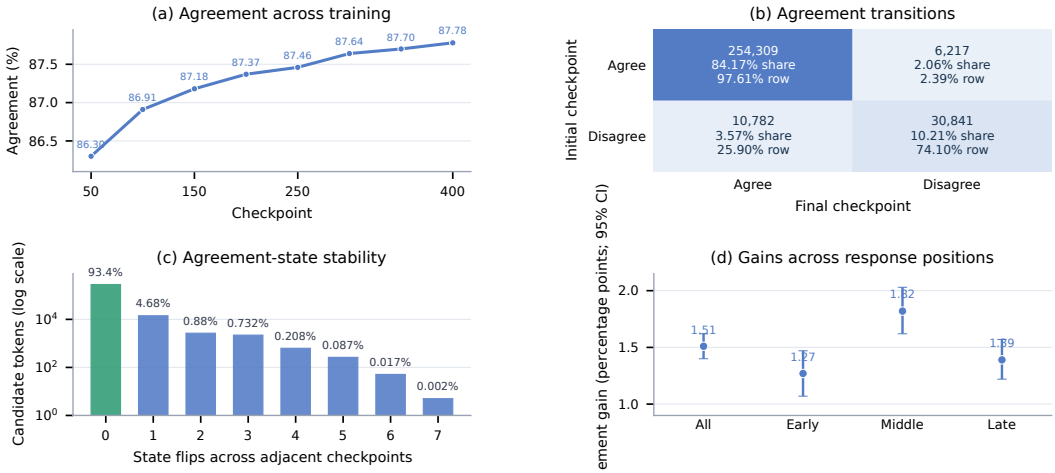

Figure 8: **Layer agreement across training.** (a) Agreement on fixed candidate-token sets. (b) First-to-last transitions; each cell reports count, overall share, and row-conditional rate. (c) Full flip-count distribution; labels report candidate-token shares. (d) Agreement gains across response positions; error bars are question-cluster bootstrap 95% CIs.

Increasing agreement and net conflict resolution. Figure 8(a) shows increasing agreement at every observed checkpoint. Figure 8(b) identifies the source of this change from step 50 to 400: 10,782 candidate tokens move from conflict to agreement, compared with 6,217 in the reverse direction. Most tokens retain their initial state, so the increase reflects net conflict resolution rather than widespread reassignment.

Stability across the observed trajectory. Figure 8(c) reports all flip counts on the probability common set. Of the 281,586 candidate tokens with no flips, 252,241 agree throughout and 29,345 disagree throughout. Only 5,803 candidate tokens (1.92%) flip at least twice. Thus, the net improvement is accompanied by predominantly stable agreement states across the observed training checkpoints.

Coverage across positions. Figure 8(d) checks whether the trend is confined to one part of a response. Early, middle, and late segments cover positions 5%–30%, 35%–65%, and 70%–95% of each response, respectively. Agreement rises in every segment, with the largest gain in the middle. Table 12 reports the exact values and intervals.

I Shallow-Layer Sensitivity: Full Results

Hyperparameter-selection protocol. Before training, we randomly reserve a fixed problem-level subset of DAPO-Math-17K and exclude its examples from optimization. We choose the readout depth, support size K, and initial correction scale γ_0 using aggregate exact-match accuracy on this split, then fix these defaults for the final benchmark evaluation. The tables below show how performance varies around those fixed choices.

Table 12: Agreement across response segments. Agreement is in percent; gains and 95% CIs are in percentage points.

Positions	n	Initial	Final	Gain	95% CI
All	302,149	86.22	87.74	1.51	$[1.40, 1.62]$
Early	95,340	86.78	88.05	1.27	$[1.07, 1.47]$
Middle	111,360	85.38	87.20	1.82	$[1.62, 2.03]$
Late	95,449	86.65	88.04	1.39	$[1.22, 1.57]$

Depth sweep. We compare shared intermediate indices $m_T = m_S = m$ on Qwen2.5-7B-Instruct, Qwen3-0.6B, and Llama-3.1-8B-Instruct. Relative depth is m/L_S, using each student's layer count. The Qwen models use $m \in \{4, 7, 10, 13, 16, 19, 20, 21\}$, while Llama uses $m \in \{5, 8, 10, 14, 18, 21, 24\}$. Top-$K$ OPD is retained as a table reference and excluded from the depth curves.

Table 13: Qwen2.5-7B-Instruct layer sensitivity (%). Avg: six-benchmark mean.

m	$m/28$	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Top-K OPD		23.300	26.700	50.000	80.400	34.200	43.900	43.083
4	0.143	23.300	20.000	65.000	82.400	32.700	45.900	44.883
7	0.250	26.700	20.000	67.500	82.000	34.900	46.400	46.250
10	0.357	23.300	23.300	65.000	81.400	34.100	45.700	45.467
13	0.464	23.000	23.300	62.500	82.600	35.300	44.600	45.217
16	0.571	16.700	13.300	52.500	83.000	33.800	45.500	40.800
19	0.679	13.300	23.300	57.500	80.600	37.100	45.800	42.933
20	0.714	23.300	23.300	57.500	80.000	35.400	43.900	43.900
21	0.750	20.000	16.700	55.000	82.400	33.800	43.200	41.850

Table 14: Qwen3-0.6B layer sensitivity (%). Avg: six-benchmark mean.

m	$m/28$	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Top-K OPD		19.300	25.600	57.500	79.200	21.897	43.259	41.126
4	0.143	16.667	20.000	55.000	75.600	20.100	40.000	37.895
7	0.250	20.300	26.670	62.500	79.600	23.426	45.037	42.922
10	0.357	16.630	23.000	60.500	78.600	24.400	46.300	41.572
13	0.464	20.300	23.000	60.500	77.800	25.000	43.500	41.683
16	0.571	20.000	16.667	57.500	75.200	22.691	42.407	39.078
19	0.679	16.600	20.000	55.500	75.800	23.324	39.444	38.445
20	0.714	20.000	16.600	53.400	74.300	23.100	38.400	37.633
21	0.750	20.000	20.000	54.500	73.800	23.000	37.900	38.200

Table 15: Llama-3.1-8B-Instruct layer sensitivity (%). Avg: six-benchmark mean.

m	$m/32$	AIME24	AIME25	AMC23	MATH500	Minerva	OlympiadBench	Avg
Top-K OPD	–	16.666	10.000	50.000	58.000	20.279	30.111	30.843
5	0.156	13.333	16.667	55.000	66.200	24.265	30.815	34.380
8	0.250	16.667	13.330	57.500	68.200	23.897	35.852	35.908
10	0.313	16.670	16.667	55.500	66.200	23.324	34.852	35.536
14	0.438	13.333	13.330	52.500	61.200	25.162	35.333	33.476
18	0.563	13.333	10.000	52.500	63.800	25.000	31.407	32.673
21	0.656	10.000	10.000	50.500	60.800	23.410	31.331	31.007
24	0.750	10.000	13.333	45.500	59.600	21.691	29.926	30.008

A common early optimum with depth-dependent variation. Tables 13, 14, and 15 show that the strongest averages occur at relative depths near one quarter for all three students. Across tested

depths, later readouts generally reduce the average, although the exact optimum varies with model depth. This supports an early intermediate readout as a practical default while indicating model-dependent sensitivity away from that region.

J Component Ablations and Candidate-Token Pool Sources

Target reformulation without position weighting. Table 16 reports LaTR with $w_t = 1$ while retaining layer-guided target reformulation. The gain over Top-K OPD isolates target reformulation, while the difference from full LaTR measures the contribution of position weighting.

Table 16: LaTR without position weighting (%). Avg is the unweighted mean over the six math benchmarks.

Model	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Qwen2.5-7B-Instruct	23.33	23.33	65.21	82.00	34.56	45.52	45.66
Qwen3-0.6B	20.00	23.33	52.50	75.60	25.06	39.26	39.29
Qwen3-1.7B	40.30	30.30	80.50	83.40	37.20	53.70	54.23

Component contributions. Table 17 separates two operations within target reformulation across student, teacher, and union candidate-token pools. Selection adjustment chooses the final Top-K candidate tokens using layer-adjusted teacher scores, while target reweighting changes their relative teacher probabilities. With both enabled, each single-model pool contains that model's Top-128 tokens, and union combines the two sets without duplicates. The final supervision support contains 32 candidate tokens. Disabling target reweighting retains selection adjustment but renormalizes the original teacher probabilities. Disabling selection adjustment keeps the original support and only changes its target probabilities.

Table 17: Component ablations by pool (%). ✓/×: enabled/disabled. Avg: six-benchmark mean. Bold: column best within each pool.

Pool	Selection adjustment	Target reweighting	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Student	✓	✓	**23.30**	23.30	**57.50**	**82.20**	32.30	**44.70**	**43.88**
	✓	×	16.66	**23.33**	52.50	79.20	31.90	42.90	41.08
	×	✓	20.00	20.00	50.50	78.20	**32.90**	40.90	40.42
Teacher	✓	✓	**26.70**	**23.30**	**60.00**	**83.80**	**34.50**	**45.20**	**45.58**
	✓	×	16.60	20.00	57.50	80.00	33.08	42.96	41.69
	×	✓	16.70	20.00	52.50	76.20	32.00	41.70	39.85
Union	✓	✓	**26.70**	20.00	**62.50**	**81.40**	**35.70**	**45.50**	**45.30**
	✓	×	23.30	**26.60**	47.50	80.20	33.45	43.25	42.38
	×	✓	23.30	23.30	50.50	78.20	33.60	42.50	41.90

Candidate-token pool source. We examine how the source of candidate tokens affects LaTR with selection adjustment and target reweighting both enabled. The teacher Top-P pool is the default, while the student and union pools are sensitivity variants that broaden the candidate-token source. At each prefix, the *student* and *teacher* pools contain the respective model's highest-probability P tokens. The *union* pool combines these sets, counting shared tokens once. LaTR then selects K candidate tokens and constructs their target probabilities. Figure 9 shows that all three pools improve on Top-K OPD (43.08%): student, teacher, and union reach

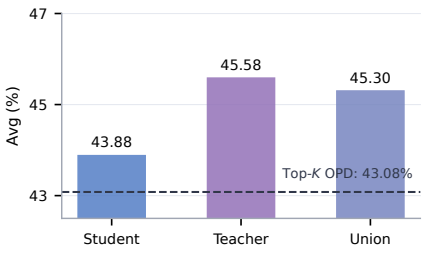

Figure 9: Candidate-token pool source. The dashed line marks Top-K OPD.

43.88%, 45.58%, and 45.30%, respectively. Pools containing teacher-preferred tokens perform better than the student pool, while adding student tokens to the teacher pool brings no further average gain.

Complementary gains across pool sources. Enabling both operations gives the highest observed six-benchmark average for every pool. Compared with selection adjustment alone, it gains 2.80, 3.89, and 2.92 points for the student, teacher, and union pools, respectively. The corresponding gains over target reweighting alone are 3.46, 5.73, and 3.40 points. Both operations therefore contribute across candidate-token sources. With both enabled, all three pools outperform Top-K OPD, and the default teacher pool achieves the highest average.

K Layer-Evidence Controls and Mechanism Ablations

The main ablations separate support reselection, target reweighting, and position weighting. Here we isolate the source and structure of the layer signal itself. The first experiment tests whether the gain requires actual intermediate-layer readouts with the correct candidate-token correspondence. The second intervenes directly on the two terms in Eq. 3 to determine which agreement relations produce the gain.

Common setup. All variants use Qwen2.5-7B-Instruct as the student, OpenThinker3-7B as the frozen teacher, DAPO-Math-17K for training, and the six math benchmarks in Section 4.1. We train for 400 updates with $P = 128$, $K = 32$, $m_T = m_S = 7$, $\gamma_0 = 1$, and linear correction decay. The candidate pool, rollout budget, optimizer, seeds, and evaluation procedure are held fixed. Table 19 changes only the indicated term in the layer-evidence equation.

Evidence-source controls. *Uniform* replaces each true intermediate distribution by the uniform distribution on the same candidate pool while retaining the final predictions. *Shuffled* preserves the measured layer evidence but randomly permutes its candidate-token assignments. *Real* uses the measured intermediate readouts with their original token identities.

Table 18: **Source of target-reformulation evidence.** Avg is the six-benchmark pass@1 mean.

Variant	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Top-K OPD	23.30	26.70	50.00	80.40	34.20	43.90	43.08
Uniform	23.30	26.70	55.00	80.20	34.60	47.10	44.48
Shuffled	23.30	16.70	55.00	82.20	33.10	44.40	42.45
Real	26.70	23.00	67.50	82.00	34.90	46.40	**46.75**

Evidence-source results. Uniform reaches 44.48%, only 1.40 points above Top-K OPD, so replacing the intermediate readouts with a token-agnostic reference provides little benefit. Shuffling the measured evidence across candidate tokens reduces the average to 42.45%, 0.63 points below Top-K OPD, showing that evidence assigned to the wrong token is harmful. Real readouts preserve both the layer signal and its token identity, reaching 46.75% and improving by 3.67 points. The comparison shows that the gain depends on token-aligned intermediate evidence rather than the presence of bounded score adjustments alone.

Mechanism controls. Let

$$r_t(v) = \underbrace{\min([u_T(v)]_+, [u_S(v)]_+)}_{\text{joint support}} - \underbrace{[-u_T(v)]_+}_{\text{teacher conflict}}.$$

No joint support removes the first term, so mutually supported corrections receive no reinforcement. *No teacher conflict* removes the second term, so corrections contradicted by the teacher's own layer evolution are not suppressed. *Random sign* preserves the absolute normalized correction magnitudes but permutes their signs across candidate tokens. The full method retains both terms and their original token assignments.

Mechanism results. The results show complementary roles for the two terms. Removing joint support loses 1.52 points, indicating that corrections endorsed by both models should be strengthened.

Table 19: **Direct ablations of the layer-agreement rule.** All settings retain the same candidate pool, correction schedule, target objective, and position weighting.

Variant	Joint support	Teacher conflict	Correct token sign	Avg	Δ from full
No joint support	×	✓	✓	44.73	−1.52
No teacher conflict	✓	×	✓	45.16	−1.09
Random sign	✓	✓	×	43.54	−2.71
Full LaTR	✓	✓	✓	**46.25**	–

Removing teacher conflict loses 1.09 points, indicating that the teacher's own layer trajectory helps identify guidance that should be attenuated. Randomizing signs causes the largest reduction and approaches the endpoint-only baseline, despite preserving the correction magnitudes. Thus the gain does not follow from adding a bounded logit perturbation of the right scale; it depends on applying the correct direction to the correct candidate token.

Conclusion. Together, the two experiments form a direct mechanism test. Actual intermediate readouts outperform uniform and shuffled controls, while interventions on Eq. 3 degrade performance. These results identify teacher–student layer agreement as the operative signal behind LaTR target reformulation. This signal captures joint support, teacher contradiction, and token-specific correction directions.

L Supervision Support and Correction Scale

The defaults in this section are selected on the held-out ablation split described in Appendix I. The final benchmark results below are reported only as post-selection sensitivity analyses.

Supervision support size. Table 20 provides all results underlying Figure 6(d). At $K = 1$, training uses a sampled-token objective based on full-vocabulary probabilities. For $K \geq 2$, both distributions are renormalized on the selected support. Average accuracy improves from 37.80% at $K = 1$ to 45.83% at $K = 32$, with a further increase of only 0.28 percentage points at $K = 64$. This confirms that the held-out choice $K = 32$ is a practical default with half as many supervised candidate tokens.

Table 20: Support-size sensitivity: six-benchmark mean accuracy (%). Gains diminish beyond $K = 32$.

K	1	8	16	32	64
Avg	37.80	39.40	44.13	45.83	46.11

Correction-scale sensitivity. After fixing $\gamma_0 = 1$ on the held-out split, we compare $\gamma_0 \in \{0, 0.5, 1, 1.5, 2\}$ on the final benchmarks, with linear decay for positive scales and no layer-based adjustment at zero. Table 21 provides the averages underlying Figure 6(c), together with the Top-K OPD reference. Linear decay from $\gamma_0 = 1$ reaches 46.25%, compared with 43.74% without adjustment. Larger and smaller scales yield lower averages, favoring a moderate initial correction.

Table 21: Correction-scale sensitivity (%).

Setting	Schedule	Avg
Top-K OPD	–	43.08
$\gamma = 0$	none	43.74
$\gamma = 0.5$	linear	45.13
$\gamma = 1$	linear	**46.25**
$\gamma = 1.5$	linear	45.20
$\gamma = 2$	linear	42.42

M Full-Vocabulary Top-K Probability-Mass Coverage

Question and protocol. We examine how much teacher probability mass a sparse Top-K support retains at student-visited prefixes. From 1,547 available Qwen2.5-7B-Instruct Top-K OPD trajectories at step 300, we sample 192 trajectories from 192 questions, stratified by the six math benchmarks. Each trajectory contributes 19 fixed response positions, from 5% to 95% in 5% increments, giving 3,648 prefixes. We evaluate OpenThinker3-7B and the step-300 student on these same prefixes. For each model X, we compute

$$\text{Mass@}\,K(h_t; X) = \sum_{v \in \text{TopK}(q_X(\cdot|h_t))} q_X(v \mid h_t), \qquad q_X(\cdot \mid h_t) = \text{softmax}(z_X^{L_X}), \tag{12}$$

using the full-vocabulary softmax normalizer at temperature 1. Each model uses its own Top-K tokens. Probabilities are not renormalized within a candidate-token pool. We report position averaged coverage and 95% percentile intervals from 2,000 bootstrap resamples clustered by question, retaining its 19 positions together.

Coverage and diminishing returns. Teacher Top-32 retains 99.72% of the full-vocabulary mass on average, leaving a 0.28% tail. Its 5th-percentile coverage is 98.59%, and source-specific means range from 99.61% on Minerva to 99.78% on AMC23. Increasing K from 16 to 32 adds 0.43 percentage points of teacher mass. Doubling it again to 64 adds only 0.15 points. The student's own Top-32 also has high mean coverage (99.33%), with a 5th percentile of 95.88%.

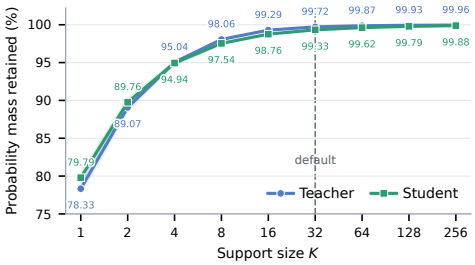

Figure 10: **Full-vocabulary Top-K probability-mass coverage** on 3,648 fixed prefixes. Shading denotes question-cluster bootstrap 95% CIs, and the dashed line marks the default $K = 32$.

Implication for sparse supervision. In this Qwen2.5 math setting, $K = 32$ retains most teacher probability mass while limiting the support used for target storage and local KL computation. Together with the small accuracy gain from $K = 32$ to 64 (Table 20), this supports our default support size. These measurements concern unadjusted supports. Appendix O establishes a separate bound on teacher-mass coverage after bounded reselection.

N Interpretation of the Four Direction-Aware Cases

The main text gives the four cases for $g_t(v) > 0$. Here we consider $g_t(v) < 0$, where the student should decrease the probability of v. The direction, layer evidence, and logit adjustment satisfy

$$s_t(v) = -1, \qquad u_X(v) = s_t(v)\Delta_X(v) = -\Delta_X(v), \qquad \eta_t(v) = \gamma s_t(v)\bar{r}_t(v) = -\gamma\bar{r}_t(v). \quad (13)$$

Thus, a model supports the required decrease exactly when its later layers weaken v. The pool-wise calibration in Eq. 4 preserves the sign of $r_t(v)$, so the four cases have the following explicit form.

Both models support the decrease $u_T > 0,\ u_S > 0$

$$\begin{aligned} \Delta_T(v) < 0,\ \Delta_S(v) < 0 \Longrightarrow u_T(v) = -\Delta_T(v) > 0, \quad u_S(v) = -\Delta_S(v) > 0, \\ A_t(v) = \min\{-\Delta_T(v), -\Delta_S(v)\}, \quad B_t(v) = 0, \\ r_t(v) = \min\{-\Delta_T(v), -\Delta_S(v)\} > 0 \Longrightarrow \eta_t(v) < 0. \end{aligned}$$

Both models weaken v. The negative adjustment therefore lowers its teacher score and unnormalized target weight.

Only the teacher supports the decrease $u_T > 0,\ u_S < 0$

$$\begin{aligned} \Delta_T(v) < 0,\ \Delta_S(v) > 0 \Longrightarrow u_T(v) > 0, \quad u_S(v) < 0, \\ A_t(v) = B_t(v) = 0 \Longrightarrow r_t(v) = 0 \Longrightarrow \eta_t(v) = 0. \end{aligned}$$

The student strengthens v, so joint support is absent. The teacher does not contradict the decrease, leaving its score unchanged.

Only the student supports the decrease $u_T < 0,\ u_S > 0$

$$\begin{aligned} \Delta_T(v) > 0,\ \Delta_S(v) < 0 \Longrightarrow u_T(v) = -\Delta_T(v) < 0, \quad u_S(v) = -\Delta_S(v) > 0, \\ A_t(v) = 0, \quad B_t(v) = \Delta_T(v), \\ r_t(v) = -\Delta_T(v) < 0 \Longrightarrow \eta_t(v) > 0. \end{aligned}$$

The teacher strengthens v, contradicting the requested decrease. The positive adjustment counteracts that decrease.

Neither model supports the decrease $u_T < 0,\ u_S < 0$

$$\Delta_T(v) > 0,\ \Delta_S(v) > 0 \Longrightarrow u_T(v) < 0, \quad u_S(v) < 0,$$
$$A_t(v) = 0, \quad B_t(v) = \Delta_T(v),$$
$$r_t(v) = -\Delta_T(v) < 0 \Longrightarrow \eta_t(v) > 0.$$

Both models strengthen v. The score is therefore raised instead of applying a decrease contradicted by the teacher's layer evolution.

O Bounds on Target Construction

We analyze bounded logit adjustments through three results. The first bounds target deviation from the teacher on the same selected support. The second isolates the resulting change in the local reverse-KL gradient at a fixed student state. The third bounds the full-vocabulary teacher mass omitted after support reselection. All three hold pointwise under their stated conditions and require no assumptions on the informativeness of the layer evidence.

Setup. Fix a prefix, a finite vocabulary $\mathcal{V}$, and a candidate-token pool $\mathcal{P} \subseteq \mathcal{V}$ of size P, and suppress the prefix index. Let $z(v) \in \mathbb{R}$ be the teacher logits over $\mathcal{V}$, and let $\eta(v)$ satisfy $|\eta(v)| \le \gamma$ for every $v \in \mathcal{P}$, where $\gamma \ge 0$. Equation 4 ensures this condition for LaTR. For $1 \le K \le P$, let $\mathcal{K}$ be the Top-K support of $z + \eta$ over $\mathcal{P}$. We use a fixed tie-breaking rule throughout.

On $\mathcal{K}$, define the teacher distribution

$$q_{\mathcal{K}}(v) = \frac{e^{z(v)}}{\sum_{u\in\mathcal{K}} e^{z(u)}}, \qquad v \in \mathcal{K}.$$

All its probabilities are positive because the logits are finite. Applying softmax to the corrected logits on this same support gives

$$p(v) = \frac{e^{z(v)+\eta(v)}}{\sum_{u\in\mathcal{K}} e^{z(u)+\eta(u)}}.$$

Dividing the numerator and denominator by $\sum_{u\in\mathcal{K}} e^{z(u)}$ expresses the target as an exponential reweighting of the teacher:

$$p(v) = \frac{q_{\mathcal{K}}(v)e^{\eta(v)}}{Z}, \qquad Z = \sum_{u\in\mathcal{K}} q_{\mathcal{K}}(u)e^{\eta(u)}, \qquad v \in \mathcal{K}. \tag{14}$$

Write $R = \max_{v\in\mathcal{K}} \eta(v) - \min_{v\in\mathcal{K}} \eta(v) \le 2\gamma$. This range captures relative adjustments, since adding a common constant to all selected logits leaves the target unchanged. In particular, $R = 0$ gives $p = q_{\mathcal{K}}$. All logarithms are natural.

Proposition O.1 (Target-deviation bound)**.** *Under the setup above,*

$$D_{\mathrm{KL}}(p\|q_{\mathcal{K}}) \le \frac{R^2}{8} \le \frac{\gamma^2}{2}. \tag{15}$$

For every $u, v \in \mathcal{K}$,

$$\log\frac{p(v)}{p(u)} - \log\frac{q_{\mathcal{K}}(v)}{q_{\mathcal{K}}(u)} = \eta(v) - \eta(u), \qquad |\eta(v) - \eta(u)| \le R \le 2\gamma. \tag{16}$$

Proof. We interpolate between the teacher and target by scaling the correction with $\beta \in [0, 1]$. Define the log normalizer and the corresponding distribution as

$$F(\beta) = \log \sum_{v\in\mathcal{K}} q_{\mathcal{K}}(v)e^{\beta\eta(v)}, \qquad p_\beta(v) = q_{\mathcal{K}}(v)e^{\beta\eta(v)-F(\beta)}.$$

Since $q_{\mathcal{K}}$ sums to one, $F(0) = 0$, $p_0 = q_{\mathcal{K}}$, $F(1) = \log Z$, and $p_1 = p$. Differentiating the finite sum gives

$$F'(\beta) = \frac{\sum_{v\in\mathcal{K}} q_{\mathcal{K}}(v)\eta(v)e^{\beta\eta(v)}}{\sum_{v\in\mathcal{K}} q_{\mathcal{K}}(v)e^{\beta\eta(v)}} = \mathbb{E}_{p_\beta}[\eta],$$
$$F''(\beta) = \mathbb{E}_{p_\beta}[\eta^2] - \big(\mathbb{E}_{p_\beta}[\eta]\big)^2 = \mathrm{Var}_{p_\beta}(\eta).$$

To bound this variance uniformly along the interpolation, let c be the midpoint of the range of η on $\mathcal{K}$. Every selected value satisfies $|\eta(v) - c| \le R/2$, so

$$F''(\beta) = \mathbb{E}_{p_\beta}[(\eta - c)^2] - \big(\mathbb{E}_{p_\beta}[\eta] - c\big)^2 \le R^2/4.$$

Next, Eq. 14 gives $\log(p(v)/q_{\mathcal{K}}(v)) = \eta(v) - F(1)$. Taking its expectation under $p = p_1$ yields

$$D_{\mathrm{KL}}(p\|q_{\mathcal{K}}) = \mathbb{E}_p[\eta] - F(1) = F'(1) - F(1).$$

Integration by parts and $F(0) = 0$ then give

$$\begin{aligned}\int_0^1 \beta F''(\beta)\, d\beta &= \big[\beta F'(\beta)\big]_0^1 - \int_0^1 F'(\beta)\, d\beta \\ &= F'(1) - F(1) + F(0) = D_{\mathrm{KL}}(p\|q_{\mathcal{K}}).\end{aligned}$$

Using the variance bound in this identity proves

$$D_{\mathrm{KL}}(p\|q_{\mathcal{K}}) \le \frac{R^2}{4}\int_0^1 \beta\, d\beta = \frac{R^2}{8} \le \frac{\gamma^2}{2}.$$

For the relative probability ratio, the common normalizer cancels:

$$\frac{p(v)}{p(u)} = \frac{q_{\mathcal{K}}(v)}{q_{\mathcal{K}}(u)} e^{\eta(v)-\eta(u)}.$$

Taking logarithms and using $|\eta(v) - \eta(u)| \le R$ gives Eq. 16. □

The KL bound does not grow with support size. Its coefficient is also locally sharp over bounded exponential reweightings: take $q_{\mathcal{K}} = (1/2, 1/2)$ and $\eta = (\gamma, -\gamma)$. Then $Z = \cosh\gamma$ and $\mathbb{E}_p[\eta] = \gamma\tanh\gamma$, giving

$$D_{\mathrm{KL}}(p\|q_{\mathcal{K}}) = \gamma\tanh\gamma - \log\cosh\gamma = \gamma^2/2 + O(\gamma^4), \qquad \gamma \to 0.$$

Thus the coefficient $1/2$ cannot be reduced uniformly over this class for arbitrarily small γ.

Proposition O.2 (Local reverse-KL gradient perturbation)**.** *Let $a \in \mathbb{R}^K$ be the student logits on $\mathcal{K}$, $\pi = \mathrm{softmax}(a)$, and $w \ge 0$. Hold $\mathcal{K}$, $q_{\mathcal{K}}$, η, and w fixed when differentiating with respect to a. Define the local reverse-KL objectives without additional log-ratio clipping as*

$$\mathcal{L}_p(a) = wD_{\mathrm{KL}}(\pi\|p), \qquad \mathcal{L}_q(a) = wD_{\mathrm{KL}}(\pi\|q_{\mathcal{K}}).$$

Their gradient difference satisfies

$$\frac{\partial(\mathcal{L}_p - \mathcal{L}_q)}{\partial a(v)} = -w\pi(v)\big(\eta(v) - \mathbb{E}_\pi[\eta]\big), \qquad v \in \mathcal{K}, \tag{17}$$

$$\|\nabla_a\mathcal{L}_p - \nabla_a\mathcal{L}_q\|_1 \le \frac{wR}{2} \le w\gamma. \tag{18}$$

Proof. Subtracting the two objectives cancels the student log probabilities. Using Eq. 14 and $\sum_v \pi(v) = 1$, we obtain

$$\begin{aligned}\mathcal{L}_p - \mathcal{L}_q &= w\sum_{v\in\mathcal{K}} \pi(v)\log\frac{q_{\mathcal{K}}(v)}{p(v)} \\ &= w\sum_{v\in\mathcal{K}} \pi(v)\big(\log Z - \eta(v)\big) \\ &= w\big(\log Z - \mathbb{E}_\pi[\eta]\big).\end{aligned}$$

The target branch is fixed, so Z and η have no derivatives with respect to a. The softmax derivative is

$$\frac{\partial\pi(u)}{\partial a(v)} = \pi(u)\big(\mathbf{1}\{u = v\} - \pi(v)\big).$$

Consequently,

$$\begin{aligned}\frac{\partial(\mathcal{L}_p - \mathcal{L}_q)}{\partial a(v)} &= -w\sum_{u\in\mathcal{K}} \eta(u)\pi(u)\big(\mathbf{1}\{u = v\} - \pi(v)\big) \\ &= -w\pi(v)\big(\eta(v) - \mathbb{E}_\pi[\eta]\big),\end{aligned}$$

which proves Eq. 17. Taking the ℓ_1 norm and applying Cauchy–Schwarz gives

$$\|\nabla_a \mathcal{L}_p - \nabla_a \mathcal{L}_q\|_1 = w \sum_{v \in \mathcal{K}} \pi(v) \big|\eta(v) - \mathbb{E}_\pi[\eta]\big| \\ \leq w\sqrt{\mathrm{Var}_\pi(\eta)} \leq \frac{wR}{2} \leq w\gamma.$$

The variance bound follows from the same midpoint argument used in Proposition O.1, now with expectation under π. □

Remark O.1 (Scope of the gradient comparison). The comparison uses the same student state, corrected support, and detached position weight in both objectives. It measures only the effect of reweighting the target on that support, consistent with the fixed-supervision branch in Algorithm 1. It does not compare complete training runs with different supports or weights, and the $w\gamma$ bound is on selected-logit gradients rather than model-parameter gradients. If additional log-ratio clipping is used, the result applies where all log-ratios in both objectives lie strictly inside the clipping interval. Bounded gradient perturbation does not imply a smaller total gradient norm or convergence of parameter optimization.

Proposition O.3 (Teacher-mass coverage after reselection)**.** *Let*

$$Q(v) = \frac{e^{z(v)}}{\sum_{u \in \mathcal{V}} e^{z(u)}}, \qquad Q(A) = \sum_{v \in A} Q(v),$$

be the full-vocabulary teacher distribution and its mass on a set A. Let $\mathcal{K}_0$ be its Top-K support over $\mathcal{V}$, and suppose $\mathcal{K}_0 \subseteq \mathcal{P}$. Define $\alpha_0 = Q(\mathcal{K}_0)$ and $\alpha = Q(\mathcal{K})$. Then

$$\alpha \geq 1 - e^{2\gamma}(1 - \alpha_0). \tag{19}$$

Proof. Let $A = \mathcal{K}_0 \setminus \mathcal{K}$ contain the displaced candidate tokens and $B = \mathcal{K} \setminus \mathcal{K}_0$ the entering candidate tokens. Both supports have size K, so $|A| = |B|$. If these sets are empty, then $\alpha = \alpha_0$ and the bound holds since $e^{2\gamma} \geq 1$.

Otherwise, fix any $u \in A$ and $v \in B$. Original teacher ranking gives $z(u) \geq z(v)$. The containment assumption ensures that u was available for corrected selection. Since v is selected and u is not, $z(v) + \eta(v) \geq z(u) + \eta(u)$. Therefore

$$0 \leq z(u) - z(v) \leq \eta(v) - \eta(u) \leq 2\gamma, \qquad Q(u) \leq e^{2\gamma} Q(v).$$

Thus an entering candidate token must carry at least $e^{-2\gamma}$ times the teacher mass of a displaced candidate token. Pair the elements of A and B by any bijection and sum this inequality to obtain

$$Q(A) \leq e^{2\gamma} Q(B).$$

The mass after selection satisfies $\alpha = \alpha_0 - Q(A) + Q(B)$. Moreover, B lies outside the original teacher Top-K, so $Q(B) \leq 1 - \alpha_0$. Combining these facts gives

$$\begin{aligned} 1 - \alpha &= 1 - \alpha_0 + Q(A) - Q(B) \\ &\leq 1 - \alpha_0 + (e^{2\gamma} - 1) Q(B) \\ &\leq 1 - \alpha_0 + (e^{2\gamma} - 1)(1 - \alpha_0) \\ &= e^{2\gamma}(1 - \alpha_0), \end{aligned}$$

which rearranges to Eq. 19. All ranking inequalities are non-strict, so the argument also covers tied scores. □

The containment assumption is satisfied by a teacher Top-P pool with $P \geq K$, or any larger pool containing it. The bound concerns teacher probability mass on the selected support. It does not require the adjusted target to preserve each teacher probability. In particular, a small original tail permits probability reweighting while retaining a high-coverage support.

The three results describe distinct effects of the same bounded correction. The KL bound controls distributional deviation after normalization, and the gradient bound controls its additional local optimization signal under fixed supervision. The coverage bound separately controls the teacher mass omitted by support reselection. These statements characterize the magnitude of the intervention without requiring that closeness to the teacher alone imply better task performance.

P COMPUTATIONAL COST

Setup and measurement. We compare one Qwen3-1.7B math run per method with the same 4B teacher and a budget of 400 updates on a single node with 8 NVIDIA A100 GPUs. These cost runs use a prompt batch size of 16, 8 responses per prompt, maximum prompt/response lengths of 2,048/16,384, learning rate 2×10^{-6}, and $K = 32$. LaTR uses a teacher candidate-token pool of 128, intermediate layer 7 in both models, and initial $\gamma = 1$ with linear decay. The recorded rollout GPU-memory utilization settings are 0.6 for Top-K OPD and 0.5 for LaTR. Log-probability micro-batch sizes per GPU are 4 and 1, respectively. The comparison therefore characterizes the two recorded training configurations.

We report total logged loop time and steady-state training metrics excluding validation and checkpoint saving. Throughput and time per million tokens account for differences in processed token volume, counting non-padding prompt and response tokens.

Table 22: **Qwen3-1.7B math training cost.** Total loop time and steady-state measurements from one run per method.

Metric	Top-K OPD	LaTR	Change
Logged loop time, 400 updates (h)	45.35	50.94	$+12.32\%$
Median active update time (s)	390.82	439.47	$+12.45\%$
Active processed tokens (M)	380.07	406.83	$+7.04\%$
Processed tokens/s/GPU	360.35	341.59	-5.21%
Active seconds/M processed tokens	346.88	365.94	$+5.49\%$

Results. Table 22 shows a 12.32% increase in logged loop time at the same update budget. Across the active window, LaTR processes 7.04% more tokens. Its time per million tokens increases by 5.49%, corresponding to 5.21% lower per-GPU throughput. Together with the accuracy gains in Table 1, these measurements indicate improved distillation performance with a modest additional runtime cost in this setting.

Q COMBINING LATR TARGET REFORMULATION WITH TOKEN-SELECTION METHODS

Several OPD methods select which generated tokens receive supervision, whereas LaTR reforms the target at each selected token. We test their compatibility by retaining each method's token-selection rule and replacing its standard target with LaTR target construction. The selector only determines which student-generated positions contribute to the loss. At retained positions, LaTR constructs its layer-guided candidate-token target. To isolate target reformulation, the combined variants omit the LaTR position weight. All comparisons use Qwen2.5-7B-Instruct with OpenThinker3-7B as teacher, the DAPO-Math-17K training split, and the six math benchmarks from Section 4.1. Training budgets, rollout settings, and evaluation procedures are held fixed.

Entropy-only. This control selects student-generated tokens by predictive entropy under the same supervision budget. Selected tokens use the standard Top-K OPD target alone and the LaTR target in the combined variant.

TIP-style. TIP selects tokens using student entropy and teacher–student divergence (Xu et al., 2026). We preserve this rule and apply LaTR target construction only in the combined variant.

Teachability-Aware OPD (TA-OPD). TA-OPD selects learnable tokens using disagreement weighted by teacher probability mass on the student support (Wang et al., 2026). We preserve this rule and apply LaTR target construction only in the combined variant.

Evaluation protocol. We report pass@1 percentages on AIME24, AIME25, AMC23, MATH-500, Minerva, and OlympiadBench, together with their unweighted average. Each selector is evaluated alone and with LaTR target reformulation.

Results. LaTR target reformulation improves the six-benchmark average for every selector: from 46.40% to 47.20% with entropy-only, from 44.20% to 47.13% with TIP-style, and from 46.30% to

Table 23: Combining LaTR target reformulation with token-selection methods (%). Combined variants omit the LaTR position weight. Avg is the unweighted six-benchmark mean.

Variant	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad Bench	Avg
Entropy	26.70	23.30	65.00	83.60	34.20	45.80	46.40
Entropy + LaTR target	23.30	23.30	67.50	85.40	35.50	48.30	47.20
TIP	20.00	26.70	55.00	82.40	35.30	45.50	44.20
TIP + LaTR target	26.70	26.70	65.00	82.20	35.20	47.00	47.13
TA-OPD	33.30	23.30	57.50	83.20	34.60	45.80	46.30
TA-OPD + LaTR target	30.00	20.00	70.00	82.40	35.93	46.67	47.50

47.50% with TA-OPD. These consistent gains show that target reformulation contributes independently of token selection. TA-OPD + LaTR target performs best overall and leads on AMC23 and Minerva.

R Additional Math Results at avg@32

We additionally evaluate mathematical reasoning with avg@32, averaging correctness over 32 sampled responses per problem. Table 24 compares Vanilla and LaTR across the six math benchmarks. Avg is their unweighted mean, computed before rounding individual scores.

Table 24: **Math avg@32 results (%).** Avg averages the six benchmarks.

Method	AIME24	AIME25	AMC23	MATH-500	Minerva	OlympiadBench	Avg
Qwen2.5-7B-Instruct							
Vanilla	10.42	6.56	46.33	69.06	28.85	33.11	32.39
LaTR	23.00	23.00	65.00	82.60	32.60	46.20	45.40
Qwen3-0.6B							
Vanilla	9.48	14.38	45.55	71.49	21.51	34.62	32.84
LaTR	22.42	26.14	61.78	78.04	23.15	44.41	42.66
Qwen3-1.7B							
Vanilla	30.29	21.63	58.25	77.07	30.61	50.17	44.67
LaTR	37.81	31.69	77.18	88.17	34.55	56.61	54.33
Llama-3.1-8B-Instruct							
Vanilla	3.44	2.31	17.50	36.93	16.11	12.29	14.76
LaTR	16.70	10.00	58.10	68.40	24.20	35.90	35.55

LaTR improves all six benchmarks for each Qwen student. The six-benchmark averages rise from 32.39% to 45.40% for Qwen2.5-7B, from 32.84% to 42.66% for Qwen3-0.6B, and from 44.67% to 54.33% for Qwen3-1.7B, gains of 13.01, 9.82, and 9.66 percentage points, respectively. These results show that the improvement over the undistilled students is consistent across the three Qwen model sizes under avg@32 evaluation.

S Additional Training Dynamics in Code Generation

Setup. We extend the training-dynamics comparison to Qwen3-0.6B code generation, reporting LiveCodeBench avg@32, gradient norm, and policy entropy (Figure 11). The step axis aligns training progress to Top-K OPD.

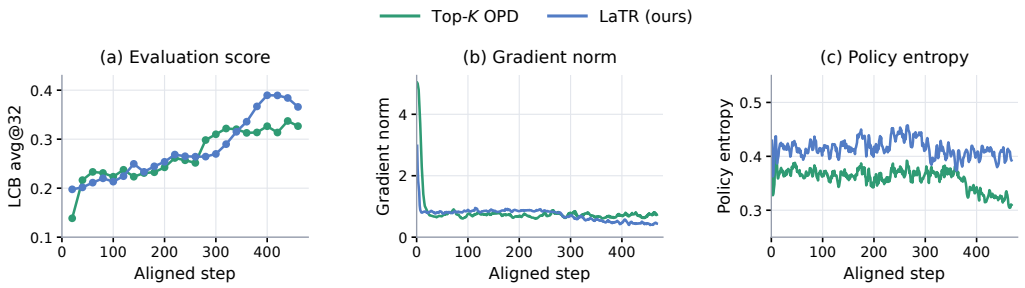

Figure 11: **Qwen3-0.6B code training dynamics.** Evaluation score, gradient norm, and policy entropy. Dashed segments, where shown, denote entropy extrapolations.

Results. LaTR improves LiveCodeBench performance over Top-K OPD, with a larger advantage late in training. Smaller initial gradient spikes suggest a less abrupt optimization start, while lower final-stage gradient norms accompany the stronger evaluation performance. Over the measured interval, its higher policy entropy is consistent with the active target-adaptation phase, in which layer evidence adjusts the relative supervision weights assigned to candidate tokens.

T Algorithm Pseudocode

Algorithm 1 summarizes LaTR training. Candidate-token operations use matching identities in $\mathcal{P}_t$ and act elementwise. Target construction is detached, so gradients flow through only the student distribution in the local reverse-KL loss.

Algorithm 1 Layer Agreement for Target Reformulation (LaTR)

Input: Frozen teacher T; student S_θ; training prompts; pool rule; $K, m_T, m_S, \delta, \epsilon$; correction schedule $\{\gamma_j\}_{j=1}^{J}$.
Output: Trained student S_θ.

for update $j = 1, \dots, J$ **do**
  Set $\gamma \leftarrow \gamma_j$, sample prompts, and generate student rollouts with valid positions $\mathcal{I}$.
  for each $t \in \mathcal{I}$ with generated prefix h_t **do**
    Compute teacher and student intermediate/final logits using the final normalization and output head; construct the shared pool $\mathcal{P}_t$.
    with gradients disabled do
      $\widetilde{q}_X^{l} \leftarrow \mathrm{softmax}(z_X^{l}[\mathcal{P}_t])$ for $X \in \{T, S\}$ and $l \in \{m_X, L_X\}$.
      $g_t \leftarrow \widetilde{q}_T^{L_T} - \widetilde{q}_S^{L_S}$; set $s_t(v) \leftarrow 0$ if $|g_t(v)| \le \delta$, and $s_t(v) \leftarrow \mathrm{sign}(g_t(v))$ otherwise.
      $\Delta_X \leftarrow \widetilde{q}_X^{L_X} - \widetilde{q}_X^{m_X}$; $u_X \leftarrow s_t \odot \Delta_X$ for $X \in \{T, S\}$.
      $r_t \leftarrow \min([u_T]_+, [u_S]_+) - [-u_T]_+$.
      $\bar{r}_t \leftarrow \mathrm{clip}\Big(r_t / (|\mathcal{P}_t|^{-1} \sum_{v \in \mathcal{P}_t} |r_t(v)| + \epsilon), -1, 1\Big)$; $\eta_t \leftarrow \gamma s_t \odot \bar{r}_t$.
      $\rho_t \leftarrow z_T^{L_T}[\mathcal{P}_t] + \eta_t$; $\mathcal{K}_t \leftarrow \mathrm{TopK}_{v \in \mathcal{P}_t}\, \rho_t(v)$.
      $p_{\mathrm{DA}} \leftarrow \mathrm{softmax}(\rho_t[\mathcal{K}_t])$; compute $w_t = \kappa_t c_t$ as defined in Section 3.2.
    end with
    $\widehat{q}_S^{L_S} \leftarrow \mathrm{softmax}(z_S^{L_S}[\mathcal{K}_t])$; $\mathcal{L}_t \leftarrow w_t D_{\mathrm{KL}}(\widehat{q}_S^{L_S} \| p_{\mathrm{DA}})$.
  end for
  Aggregate the local losses and update θ.
end for
return S_θ.

U Task Prompt Templates

We insert each example into the task-specific user prompt below, then apply the model's chat template. Braced text denotes an example-dependent field.

Math reasoning

Math problem: {*problem*}

Please carefully reason through the math problem step by step and derive the correct answer. You must conduct reasoning inside <think> and </think> and give the final answer within \boxed{}.

Code generation

You are an expert Python programmer. Solve the following competitive programming problem.
Read the input from standard input and print the answer to standard output.
Return ONLY the code inside a single ```python code block, without any explanation or tests.

Problem
{*problem*}

Sample Tests
Input:
{*sample input*}
Output:
{*sample output*}