CALIBRATING TEACHER–STUDENT DISCREPANCY FOR ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) improves reasoning models by learning the token-level discrepancy between a stronger teacher and an on-policy student. However, this discrepancy does not purely reflect the capability gap between the teacher and the student: it also contains deviations arising from the teacher itself, which are consequently mixed into the observed teacher–student discrepancy and indiscriminately learned by standard OPD during training. This issue is further exacerbated by privileged OPD, where privileged information induces larger teacher-side likelihood shifts, thereby encouraging the student to learn more of the teacher's own deviation. We introduce **Calibrated On-Policy Distillation (Cal-OPD)**, which estimates the teacher's self-deviation region through positive and negative privileged interventions and calibrates the original teacher–student discrepancy by retaining only the component that lies beyond this region. Experiments on mathematical reasoning benchmarks show that, while retaining only about 52–65% of the original teacher–student discrepancy as the optimization signal, Cal-OPD consistently outperforms standard OPD and its variants across model scales.

1 INTRODUCTION

Knowledge distillation (KD) transfers capabilities from a stronger teacher to a weaker student(Hinton et al., 2015), but conventional off-policy distillation suffers from distribution mismatch between fixed distillation data and the student's evolving policy distribution(Agarwal et al., 2024; Gu et al., 2024). On-policy distillation (OPD) mitigates this mismatch by training directly on student-generated trajectories and using teacher–student token-level likelihood discrepancies as dense supervision(Agarwal et al., 2024; Yang et al., 2025; Jin et al., 2026). Compared with reinforcement learning with verifiable rewards (RLVR), which relies on sparse outcome-level rewards(Wen et al., 2026; Guo et al., 2025; Yu et al., 2026a), OPD provides token-level guidance throughout the trajectory, enabling more direct and efficient reasoning post-training(Yang et al., 2025; Jin et al., 2026).

Standard OPD implicitly treats the teacher likelihood assigned to each student token as an equally reliable reference. Yet its stability varies substantially across tokens. Recent studies on reasoning models suggest that reasoning tokens tend to remain relatively stable, whereas stylistic or surface-form tokens, such as discourse markers and formatting choices, are substantially more sensitive to contextual interventions even when both the question and student rollout are held fixed(Pan et al., 2026; He et al., 2026). Related work further shows that the reliability of teacher supervision can vary across tokens and reasoning positions(Liu et al., 2026). We refer to this token-specific variability in teacher likelihood as **Teacher Self-Deviation (TSD)**. Consequently, the observed teacher–student discrepancy reflects not only the underlying capability gap but also deviations arising from the teacher itself, which standard OPD indiscriminately incorporates into the learning signal.

Privileged OPD extends standard OPD by conditioning a stronger teacher on additional training-time information, such as reference solutions, final answers, or hints(Ye et al., 2026; Yu et al., 2026b; Kaur et al., 2026). While such privileged context is intended to improve teacher supervision, it also induces further shifts in the teacher distribution. Existing studies show that these shifts can introduce information-asymmetry effects(Yu et al., 2026b), shortcut behavior(Tian et al., 2026), and even degrade performance in thinking models(Kaur et al., 2026). From the perspective of teacher self-deviation, privileged OPD therefore amplifies the teacher-side deviations already embedded in

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.

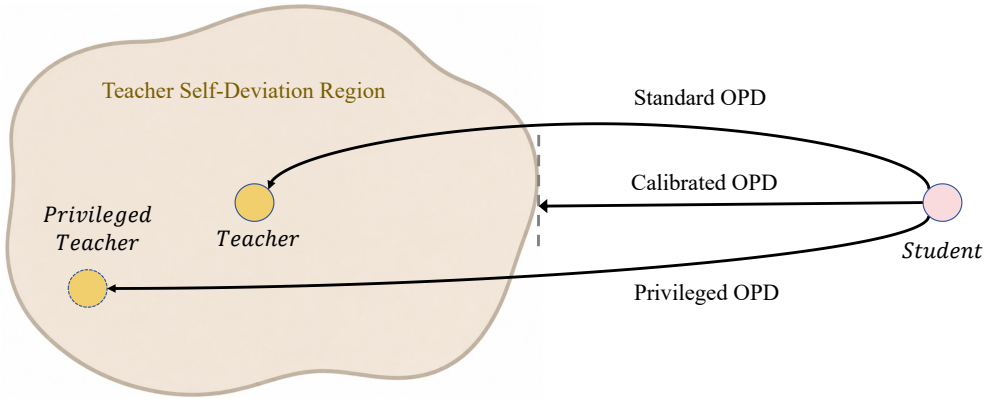


Figure 1: Conceptual comparison of standard OPD, privileged OPD, and Cal-OPD. Cal-OPD removes the TSD-explained discrepancy and retains only the residual beyond the TSD region.

the teacher–student discrepancy, causing the student to learn more context-induced teacher variation, as illustrated in Figure 1.

To address this issue, we propose **Calibrated On-Policy Distillation (Cal-OPD)**, which first estimates the teacher's self-deviation region and then learns only the teacher–student discrepancy that lies beyond it. Specifically, Cal-OPD applies positive and negative privileged interventions, whose contrasting semantics provide complementary probes of the teacher's contextual variability and enable an approximation of its token-level self-deviation region. Unlike privileged OPD, these interventions are not directly distilled into the student, but are instead used to measure and calibrate the teacher reference. Cal-OPD then removes the portion of the original discrepancy covered by the estimated self-deviation region, retaining only the residual as the learning signal and setting it to zero when the student likelihood falls within this estimated region.

Our contributions are summarized as follows:

- We identify and empirically characterize **Teacher Self-Deviation (TSD)**, showing that teacher likelihood is not an equally stable reference across tokens and contexts, and that learning these teacher-side deviations, which are further amplified by privileged context, can substantially degrade OPD for long-CoT reasoning.
- We introduce **Calibrated On-Policy Distillation (Cal-OPD)**, which uses positive and negative privileged interventions to estimate the teacher's self-deviation region and retains only the teacher–student discrepancy beyond it. Privileged information is thus used to calibrate the teacher reference rather than directly supervise the student.
- Extensive experiments on mathematical reasoning benchmarks show that Cal-OPD retains only about 52–65% of the original teacher–student discrepancy for optimization, yet consistently outperforms standard OPD and its variants across model scales, while alleviating the response-length expansion observed in standard and privileged OPD.

2 Understanding Teacher Self-Deviation

2.1 Preliminaries

On-Policy Distillation. Let $\pi_S = \pi_\theta$ denote the student policy parameterized by θ, and let π_T denote a fixed teacher policy. Given a problem $x \sim \mathcal{D}$, the student generates an on-policy rollout

$$y = (y_1, \ldots, y_T) \sim \pi_S(\cdot \mid x), \tag{1}$$

where $y_{<t} = (y_1, \ldots, y_{t-1})$ denotes the prefix of the student response preceding token y_t. For each student-generated token, we evaluate the teacher and student under the same problem x and student response prefix $y_{<t}$, and define their token-level log-likelihoods and discrepancy as

$$\ell_t^T = \log \pi_T(y_t \mid x, y_{<t}), \qquad \ell_t^S = \log \pi_S(y_t \mid x, y_{<t}), \qquad A_t^{\mathrm{OPD}} = \ell_t^T - \ell_t^S. \tag{2}$$

OPD uses A_t^{OPD} as the token-level advantage for optimizing the student:

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = -\mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_S(\cdot|x)}\left[\frac{1}{T}\sum_{t=1}^{T}\mathrm{sg}\big(A_t^{\mathrm{OPD}}\big)\log\pi_S(y_t \mid x, y_{<t})\right], \tag{3}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator. Accordingly, $A_t^{\mathrm{OPD}} > 0$ increases the student likelihood of y_t, whereas $A_t^{\mathrm{OPD}} < 0$ decreases it. This formulation treats the teacher likelihood ℓ_t^T as the token-level reference against which the student is optimized.

Teacher Self-Deviation. For a fixed problem x, student rollout y, and token position t, we introduce a teacher-side contextual intervention c while keeping the evaluated student trajectory unchanged. The intervention is provided only to the teacher, together with the problem x, and precedes the student rollout prefix $y_{<t}$ in the teacher context. We use $c_0 = \varnothing$ to denote the original setting without additional context. The teacher likelihoods under intervention c and the original setting c_0, together with the induced likelihood variation, are defined as

$$\begin{gathered}\ell_t^T(c) = \log\pi_T(y_t \mid x, c, y_{<t}), \qquad \ell_t^T(c_0) = \log\pi_T(y_t \mid x, y_{<t}) = \ell_t^T, \\ \Delta_t^T(c) = \ell_t^T(c) - \ell_t^T(c_0).\end{gathered} \tag{4}$$

Since x, $y_{<t}$, and y_t remain fixed throughout the comparison, $\Delta_t^T(c)$ captures the change in teacher likelihood induced by the additional context c, rather than any change in the evaluated trajectory. We refer to this context-induced variation in teacher likelihood as **teacher self-deviation (TSD)**.

Given an intervention set $\mathcal{C}$, we use the induced likelihood variations to estimate the downward and upward magnitudes of TSD:

$$\hat{d}_t^{\downarrow} = \max\left(0, -\min_{c\in\mathcal{C}}\Delta_t^T(c)\right), \qquad \hat{d}_t^{\uparrow} = \max\left(0, \max_{c\in\mathcal{C}}\Delta_t^T(c)\right). \tag{5}$$

Both $\hat{d}_t^{\downarrow}$ and $\hat{d}_t^{\uparrow}$ are non-negative and measure the maximum downward and upward deviations in teacher log-likelihood, respectively. These quantities yield an empirical estimate of the **TSD region**:

$$\hat{\mathcal{R}}_t^T(\mathcal{C}) = \left[\ell_t^T - \hat{d}_t^{\downarrow},\ \ell_t^T + \hat{d}_t^{\uparrow}\right]. \tag{6}$$

Accordingly, $\hat{\mathcal{R}}_t^T(\mathcal{C})$ is a finite-intervention approximation to the underlying teacher self-deviation region $\mathcal{R}_t^T$, rather than an exhaustive characterization of all possible contextual variation. TSD characterizes variability in the teacher likelihood, while student-generated rollouts provide the trajectories on which this variability is measured.

2.2 Teacher Self-Deviation Is Not Reliable Knowledge

TSD captures contextual changes in the teacher likelihood assigned to a fixed token. If TSD **faithfully reflected task-relevant knowledge**, its variation should be systematically tied to the information introduced by the context. In particular, TSD should **depend on the presence of task-specific information** and **respond consistently to the correctness of that information**. Otherwise, the observed likelihood shift cannot be reliably interpreted as a knowledge signal. Table 1 summarizes the contextual interventions used in our analysis, spanning task-agnostic instructions, evaluative feedback, and answer- and solution-level privileged information. We conduct this analysis with Qwen3-1.7B as the student and Qwen3-8B(Yang et al., 2025) as the teacher, both in thinking mode, over 6,528 questions sampled from DAPO-17k(Yu et al., 2026a), with one response per question and approximately 60 million response tokens evaluated across all intervention conditions. Details of data collection and preparation for TSD analysis are provided in Appendix A.2.

TSD Emerges Without Task Knowledge. We first find that **substantial TSD emerges even in the absence of external task-specific knowledge**, and that the affected token positions are largely preserved when richer privileged information is subsequently introduced. This indicates that part of the teacher-side variability amplified by privileged context is already present before answer- or solution-level knowledge is provided.

To quantify this effect, we group the positive and negative variants of each intervention group as

$$\mathcal{C}_g = \left\{c_g^{\mathrm{pos}}, c_g^{\mathrm{neg}}\right\}, \qquad g \in \{\mathrm{inst}, \mathrm{eval}, \mathrm{ans}, \mathrm{sol}\}. \tag{7}$$

Table 1: Contextual interventions used to probe TSD. The shared problem context x is omitted for brevity. For answer- and solution-level interventions, the positive and negative variants share the same template and differ only in the inserted privileged content.

Symbol	Contextual Intervention
	Task-Agnostic Instructions
$c_{\text{inst}}^{\text{pos}}$	Please reason through the problem **carefully and thoroughly**. **Verify intermediate steps** and provide a complete, rigorous solution.
$c_{\text{inst}}^{\text{neg}}$	Please solve the problem **quickly and directly**. **Avoid unnecessary elaboration or extensive verification** and reach the final answer efficiently.
	Evaluative Feedback
$c_{\text{eval}}^{\text{pos}}$	A gold-standard verifier has judged that the following solution **reaches the correct final answer**. The reasoning is **rigorous, coherent, and mathematically sound**.
$c_{\text{eval}}^{\text{neg}}$	A gold-standard verifier has judged that the following solution **does not reach the correct final answer**. The reasoning is **flawed, incoherent, and mathematically unreliable**.
	Answer-Level Privilege
$c_{\text{ans}}^{\text{pos/neg}}$	A verified ground-truth answer is provided as a reliable reference: ***[correct / incorrect answer]***. Use it to guide your reasoning while providing a complete and logically coherent solution.
	Solution-Level Privilege
$c_{\text{sol}}^{\text{pos/neg}}$	A reference solution is provided as additional guidance: ***[correct / unrelated solution]***. Use it while independently providing a complete and logically coherent solution.

Given a threshold τ, we define the set of tokens exhibiting **significant TSD** under group $\mathcal{C}_g$ as

$$\mathcal{S}_g(\tau) = \left\{ t : \max_{c \in \mathcal{C}_g} \left| \Delta_t^T(c) \right| > \tau \right\}. \tag{8}$$

Figure 2(a) reports the prevalence of significant TSD under each intervention, together with the union prevalence defined by $\mathcal{S}_g(\tau)$. At $\tau = 0.01$, task-agnostic instructions induce significant TSD on 20.2% and 25.8% of tokens under the positive and negative variants, respectively, with their union covering 29.8% of all tokens. This prevalence is comparable to the 29.1% observed under evaluative feedback and the 29.4% under answer-level privilege, despite task-agnostic interventions providing neither an answer nor a solution. Solution-level privilege further expands the affected set to 39.6%, showing that richer privileged context broadens TSD rather than creating it from scratch.

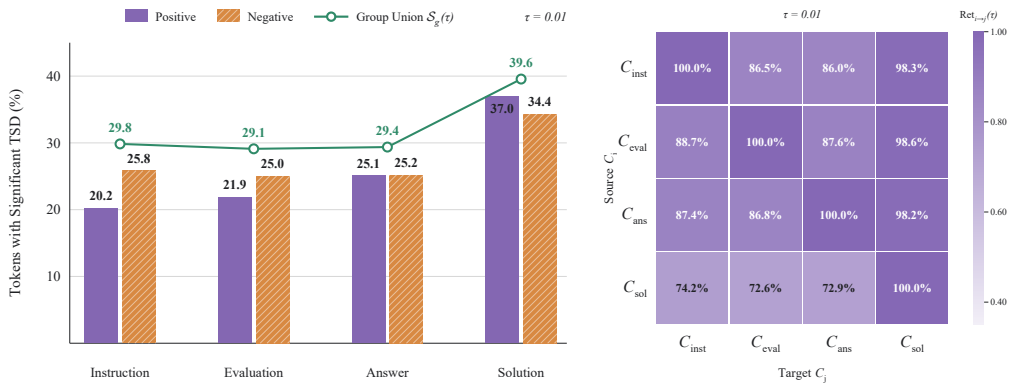
We measure whether TSD under less informative interventions persists under richer privileged contexts. For groups $\mathcal{C}_i$ and $\mathcal{C}_j$, we define the retention of significant TSD as

$$\text{Ret}_{i \to j}(\tau) = \frac{|\mathcal{S}_i(\tau) \cap \mathcal{S}_j(\tau)|}{|\mathcal{S}_i(\tau)|}. \tag{9}$$

Figure 2(b) reveals a clear asymmetry. Under solution-level privilege, 98.3%, 98.6%, and 98.2% of tokens exhibiting significant TSD under task-agnostic instructions, evaluative feedback, and answer-level privilege remain significant, respectively, whereas only 74.2%, 72.6%, and 72.9% of solution-level significant-TSD tokens remain significant under these less informative interventions. Thus, richer privileged context largely preserves previously affected positions while extending TSD to additional positions. Together, these results show that **task knowledge is not necessary for TSD to emerge**, while richer privileged contexts mainly broaden the affected set rather than introducing an entirely new deviation pattern.

TSD Is Largely Insensitive to Intervention Semantics. We find that TSD is largely **insensitive to both semantic polarity and correctness**. To quantify this consistency, we compare the positive and negative variants of evaluative, answer-level, and solution-level interventions using three metrics. For intervention group g, let $\Delta_t^+ = \Delta_t^T(c_g^{\text{pos}})$ and $\Delta_t^- = \Delta_t^T(c_g^{\text{neg}})$, with corresponding significant-TSD sets $S_g^+(\tau) = \{t : |\Delta_t^+| > \tau\}$ and $S_g^-(\tau) = \{t : |\Delta_t^-| > \tau\}$. **Overlap** measures whether significant TSD occurs at the same token positions, while **Agreement** measures whether the two interventions shift teacher likelihood in the same direction at jointly significant positions:

$$\text{Ovl}_g(\tau) = \frac{|S_g^+(\tau) \cap S_g^-(\tau)|}{|S_g^+(\tau) \cup S_g^-(\tau)|}, \qquad \text{Agr}_g(\tau) = \frac{\sum_{t \in S_g^+(\tau) \cap S_g^-(\tau)} \mathbf{1}\left[\Delta_t^+ \Delta_t^- > 0\right]}{|S_g^+(\tau) \cap S_g^-(\tau)|}. \tag{10}$$



(a) Prevalence of Significant TSD **(b) Retention of Significant TSD**

Figure 2: TSD across contextual interventions. Significant TSD emerges even under task-agnostic interventions, and the affected token positions are largely retained under richer privileged contexts.

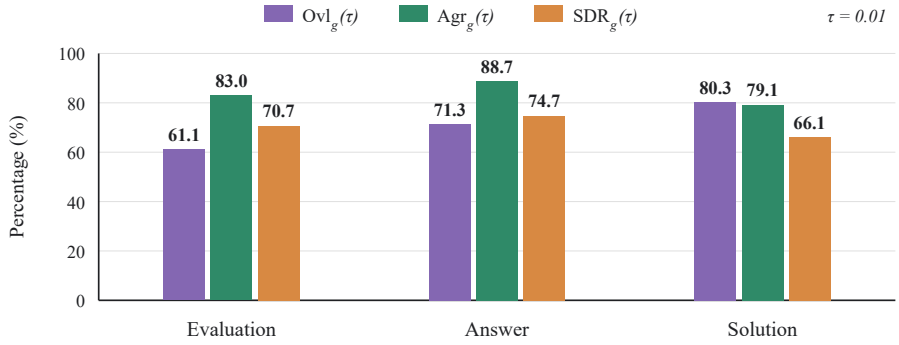


Figure 3: Consistency of TSD under contrasting interventions. TSD occurs at overlapping token positions, remains directionally aligned, and is dominated by a shared common-mode component.

We further measure how much paired variation is shared between the two interventions. Let $I_g(\tau) = S_g^+(\tau) \cap S_g^-(\tau)$ denote the set of jointly significant tokens. We decompose the paired deviations into shared and contrastive components and define the **Shared Deviation Ratio (SDR)** as

$$M_t = \frac{\Delta_t^+ + \Delta_t^-}{2}, \qquad D_t = \frac{\Delta_t^+ - \Delta_t^-}{2}, \qquad \mathrm{SDR}_g(\tau) = \frac{\sum_{t \in I_g(\tau)} |M_t|}{\sum_{t \in I_g(\tau)} (|M_t| + |D_t|)}. \tag{11}$$

A higher shared deviation ratio indicates that, among jointly significant tokens, paired TSD is dominated by variation shared across interventions, with less attributable to their semantic difference.

Figure 3 reveals that TSD remains highly structured under contrasting intervention semantics. Reversing answer correctness still yields 88.7% directional agreement and a 74.7% shared deviation ratio, showing that paired likelihood shifts are dominated by a shared response rather than the correctness contrast itself. Solution-level interventions exhibit the highest positional overlap at 80.3%, yet the lowest shared deviation ratio at 66.1%, suggesting that richer context alters how TSD varies more than where it emerges. Across all groups, substantial overlap, high directional agreement, and predominantly shared variation persist. These results show that **TSD is largely insensitive to intervention semantics and correctness**, further indicating that such teacher likelihood shifts cannot be reliably interpreted as task knowledge. Additional analyses across multiple thresholds and model scales in Appendix A.3 reproduce these findings.

2.3 TSD Concentrates on Surface-Form Tokens

TSD is **strongly concentrated on surface-form tokens rather than tokens carrying mathematical content**. High-TSD forms are natural-language markers that organize, qualify, or redirect the

Table 2: Token forms with the highest and lowest significant-TSD rates $\rho_\tau(v)$ under $c_{\text{sol}}^{\text{pos}}$ at $\tau = 0.01$, restricted to forms occurring more than 20,000 times. $\overline{|\Delta|}$ denotes the mean TSD magnitude over significant occurrences. Values are rounded for display; rankings are based on the unrounded $\rho_\tau(v)$. For compactness, `altern.` abbreviates `alternatively`.

#	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	#	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	#	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$
					Highest ρ_τ						
1	`maybe`	97.4	0.59	7	`seems`	93.6	0.41	13	`wait`	90.0	0.42
2	`however`	97.4	0.63	8	`another`	92.9	0.50	14	`check`	89.6	0.28
3	`therefore`	95.9	0.46	9	`since`	91.8	0.44	15	`let`	89.5	0.50
4	`consider`	95.4	0.38	10	`try`	91.8	0.40	16	`here`	89.4	0.36
5	`altern.`	94.3	0.58	11	`think`	91.6	0.40	17	`because`	89.4	0.40
6	`earlier`	94.2	0.41	12	`says`	90.4	0.25	18	`now`	89.2	0.49
					Lowest ρ_τ						
1	`}{`	2.8	0.25	7	`°`	7.3	0.17	13	`frac`	8.0	0.22
2	`###`	3.2	0.13	8	`_`	7.3	0.29	14	`{`	8.0	0.23
3	`0`	5.0	0.36	9	`6`	7.4	0.40	15	`4`	8.8	0.37
4	`9`	6.8	0.38	10	`7`	7.7	0.39	16	`2`	8.8	0.32
5	`8`	7.2	0.39	11	`5`	7.8	0.38	17	`3`	9.2	0.36
6	$\surd$	7.2	0.28	12	θ	7.9	0.28	18	`_i`	9.2	0.28

reasoning text without directly encoding problem-specific mathematical information, whereas low-TSD forms are dominated by numbers, mathematical symbols, and notation. Under $c_{\text{sol}}^{\text{pos}}$, we merge tokenizer variants corresponding to the same surface form and restrict the analysis to forms occurring more than 20,000 times. For each token form v, we define the significant-TSD rate as $\rho_\tau(v) = \Pr\big(|\Delta_t^T(c_{\text{sol}}^{\text{pos}})| > \tau \mid y_t = v\big)$. Unlike occurrence counts, $\rho_\tau(v)$ measures how often a token exhibits significant TSD when it appears, thereby controlling for differences in token frequency.

Table 2 reveals a clear separation between token forms with the highest and lowest significant-TSD rates at $\tau = 0.01$. The 18 highest-ranked forms are dominated by natural-language surface expressions such as `maybe`, `however`, `therefore`, `consider`, and `alternatively`, with ρ_τ exceeding 89% throughout, meaning significant TSD appears in about nine of ten occurrences. In contrast, the 18 lowest-ranked forms consist predominantly of digits, mathematical symbols, and notation such as `0`, $\surd$, θ, and `frac`, all with ρ_τ below 9.3%. This nearly order-of-magnitude separation shows that significant TSD occurs far more frequently on surface-form tokens than on tokens directly expressing mathematical content. The large $\overline{|\Delta|}$ values in Table 2 mainly reflect the stronger teacher-side shifts induced by solution-level privilege. Appendix A.4 extends the analysis to the top-24 and bottom-24 token forms across different contextual interventions, showing that weaker interventions substantially reduce deviation magnitudes while preserving the same separation between surface-form tokens and mathematical or symbolic forms. Appendix A.5 provides a trajectory-level case study illustrating how TSD is distributed throughout a complete reasoning trace.

3 Calibrated On-Policy Distillation

The preceding analysis shows that teacher likelihood is not an equally reliable pointwise reference: it exhibits substantial TSD that cannot be reliably interpreted as a knowledge signal. We therefore propose **Calibrated On-Policy Distillation (Cal-OPD)**, which decomposes the observed teacher–student discrepancy into a TSD-explained component and a calibrated residual, and distills only the discrepancy that remains beyond the teacher's estimated self-deviation region.

$$\underbrace{A_t^{\text{OPD}}}_{\text{Teacher–Student Discrepancy}} = \underbrace{A_t^{\text{Cal}}}_{\text{Calibrated Teacher–Student Discrepancy}} + \underbrace{A_t^{\text{TSD}}}_{\text{TSD-Explained Discrepancy}}. \tag{12}$$

For each token y_t, Cal-OPD probes the teacher with two contrasting interventions c^{pos} and c^{neg}, producing $\Delta_t^{\text{pos}} = \Delta_t^T(c^{\text{pos}})$ and $\Delta_t^{\text{neg}} = \Delta_t^T(c^{\text{neg}})$. We define their maximum downward and

Table 3: Main results on mathematical reasoning benchmarks. The best result among distillation methods for each teacher–student configuration is shown in **bold**. Subscripts on Cal-OPD indicate gains over the corresponding student. For compactness, **4B-2507** and **30B-2507** denote Qwen3-4B-Thinking-2507 and Qwen3-30B-A3B-Thinking-2507, respectively.

Model / Method	AMC23	AIME24	AIME25	AIME26	HMMT26	MATH500	Avg.
4B-2507 → Qwen3-1.7B							
Student	80.8	40.6	32.5	29.2	21.8	90.2	49.2
Teacher	94.8	56.5	53.1	54.2	25.2	95.4	63.2
OPD	82.0	39.6	35.0	34.6	22.9	90.6	50.8
ExOPD	81.7	43.1	35.4	34.2	**25.6**	90.6	51.8
EOPD	81.6	44.0	36.7	**38.8**	19.7	90.6	51.9
Uni-OPD	81.4	42.1	34.4	32.7	22.5	89.8	50.5
Privileged-OPD	83.3	39.8	30.4	31.0	21.8	89.4	49.3
Cal-OPD	$\mathbf{84.5}_{+3.7}$	$\mathbf{45.0}_{+4.4}$	$\mathbf{37.1}_{+4.6}$	$36.3_{+7.1}$	$24.8_{+3.0}$	$\mathbf{90.8}_{+0.6}$	$\mathbf{53.1}_{+3.9}$
30B-2507 → Qwen3-4B							
Student	95.2	64.8	56.3	57.9	30.1	95.4	66.6
Teacher	97.8	74.8	65.4	66.0	33.9	96.9	72.5
OPD	94.2	62.1	57.5	55.2	31.1	95.0	65.9
ExOPD	92.0	66.9	59.4	56.3	**33.7**	95.6	67.3
EOPD	95.3	62.7	54.6	60.0	33.5	**96.7**	67.1
Uni-OPD	93.4	**67.9**	59.8	57.9	31.4	95.4	67.6
Privileged-OPD	93.4	58.8	52.1	54.0	31.3	94.4	64.0
Cal-OPD	$\mathbf{96.3}_{+1.1}$	$67.1_{+2.3}$	$\mathbf{61.3}_{+5.0}$	$\mathbf{60.2}_{+2.3}$	$33.3_{+3.2}$	$95.8_{+0.4}$	$\mathbf{69.0}_{+2.4}$

upward deviations as $\hat{d}_t^{\downarrow} = \max(0, -\Delta_t^{\mathrm{pos}}, -\Delta_t^{\mathrm{neg}})$ and $\hat{d}_t^{\uparrow} = \max(0, \Delta_t^{\mathrm{pos}}, \Delta_t^{\mathrm{neg}})$. Since two interventions provide only a finite probe of the underlying TSD, we introduce a relaxation factor $\lambda \geq 1$ and estimate the TSD region as

$$\hat{\mathcal{R}}_t^T = \left[\ell_t^T - \lambda \hat{d}_t^{\downarrow},\ \ell_t^T + \lambda \hat{d}_t^{\uparrow}\right] = [L_t^T, U_t^T]. \tag{13}$$

Cal-OPD then removes the portion of the teacher–student discrepancy covered by the estimated TSD region and retains only the residual beyond its boundary:

$$A_t^{\mathrm{Cal}} = \left[L_t^T - \ell_t^S\right]_+ - \left[\ell_t^S - U_t^T\right]_+, \qquad [z]_+ = \max(z, 0). \tag{14}$$

When $\ell_t^S \in \hat{\mathcal{R}}_t^T$, the observed teacher–student discrepancy is fully covered by the estimated TSD region and $A_t^{\mathrm{Cal}} = 0$; otherwise, A_t^{Cal} retains only the discrepancy beyond the nearest boundary. Accordingly, $A_t^{\mathrm{TSD}} = A_t^{\mathrm{OPD}} - A_t^{\mathrm{Cal}}$. Cal-OPD follows the standard OPD objective in Eq. 3, replacing A_t^{OPD} with A_t^{Cal} as the token-level advantage.

4 Experiments

4.1 Experimental Setup

Datasets. We use DAPO-17K (Yu et al., 2026a) filtered by Qwen3-235B-A22B-Instruct-2507(Yang et al., 2025) as the training dataset. The filtering and solution-annotation procedure is described in Appendix A.2. For evaluation, we consider six reasoning benchmarks: AMC23, AIME24, AIME25, AIME26, HMMT26, and MATH500 (Lightman et al., 2024). These benchmarks span a range of difficulty, from standard problem solving to high-difficulty competition mathematics.

Models and Baselines. We evaluate Cal-OPD on two teacher–student configurations from the Qwen3 family (Yang et al., 2025): Qwen3-4B-Thinking-2507→Qwen3-1.7B and Qwen3-30B-A3B-Thinking-2507→Qwen3-4B, covering two distinct scale regimes. We compare against five representative OPD baselines: standard OPD (Agarwal et al., 2024), which directly optimizes teacher–student discrepancies; ExOPD (Yang et al., 2026a), augmented with reward extrapolation;

Table 4: Effect of contextual interventions used for TSD estimation on Cal-OPD performance for Qwen3-4B-Thinking-2507→Qwen3-1.7B.

Intervention Set	AMC23	AIME24	AIME25	AIME26	HMMT26	MATH500	Avg.
$\mathcal{C}_{\text{inst}}$	83.9	40.8	34.8	34.2	**25.2**	90.3	51.5
$\mathcal{C}_{\text{eval}}$	**84.5**	**45.0**	**37.1**	36.3	24.8	90.8	**53.1**
$\mathcal{C}_{\text{ans}}$	84.2	40.8	34.8	**37.3**	23.9	**90.9**	52.0
$\mathcal{C}_{\text{sol}}$	81.3	40.6	30.0	30.8	22.0	89.2	49.0

EOPD (Jin et al., 2026), with entropy-aware forward-KL supervision; Uni-OPD (Hou et al., 2026a), with exploration and outcome-guided calibration; and Privileged-OPD (Ye et al., 2026; Kaur et al., 2026), where the teacher is additionally conditioned on the ground-truth reference solution. This evaluates if Cal-OPD remains effective across varying capacity gaps and diverse modifications of the standard OPD objective. We report Avg@16 accuracy, computed by sampling 16 independent responses per problem and averaging their binary correctness scores across the full evaluation set.

Implementation Details. We implement all methods with `verl` (Sheng et al., 2025) and train on 8 NVIDIA H20 GPUs, with 4 GPUs hosting the student and 4 hosting the teacher. All methods are trained for 100 steps, with 256 trajectories per step, one rollout per question unless otherwise specified, and a learning rate of 1×10^{-6}. During training, we set response length to 16,384, temperature to 1.0, and top-p to 1.0; during evaluation, we use response length of 20,480, temperature 0.6, and top-p to 0.95. For Cal-OPD, we use $c^{\text{pos}}_{\text{eval}}$ and $c^{\text{neg}}_{\text{eval}}$ as contrasting interventions and set the relaxation factor to $\lambda = 5$. Method-specific hyperparameters are provided in Appendix A.6.

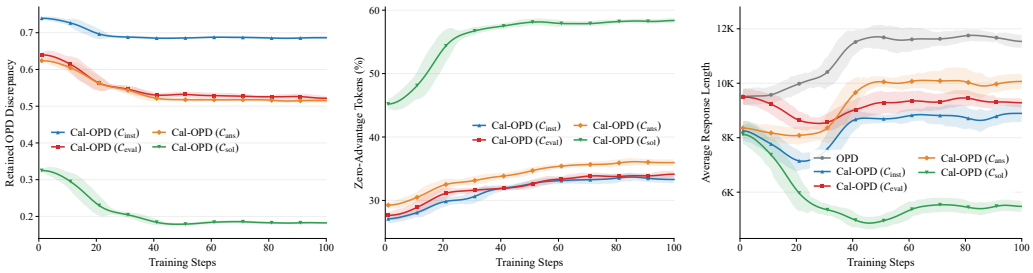
4.2 Main Results

Table 3 shows that Cal-OPD achieves the highest average performance in both teacher–student configurations: 53.1 for Qwen3-4B-Thinking-2507→Qwen3-1.7B and 69.0 for Qwen3-30B-A3B-Thinking-2507→Qwen3-4B. This corresponds to gains of $+3.9$ and $+2.4$ over the student baselines, and $+2.3$ and $+3.1$ over standard OPD. Cal-OPD also achieves the best result on 7 of 12 benchmark–configuration pairs, demonstrating consistent gains across model scales and reasoning benchmarks. In contrast, standard OPD improves the 4B→1.7B student only modestly, from 49.2 to 50.8, and even degrades the 30B→4B student from 66.6 to 65.9, despite both teachers being substantially stronger than their students. This shows that raw teacher–student discrepancy is **not uniformly beneficial supervision**: optimizing all discrepancies also learns teacher-side deviations, whereas Cal-OPD removes the TSD-explained component and retains a more effective signal.

Privileged-OPD exhibits the strongest degradation, obtaining the lowest average performance among distillation methods in both configurations, at 49.3 and 64.0, respectively. In the 4B→1.7B setting, it nearly eliminates the gain from standard OPD, while in the 30B→4B setting it falls 2.6 points below the student. Together with our earlier finding that privileged context substantially amplifies TSD, these results suggest that directly distilling the privileged teacher can transfer teacher-side deviation alongside task-relevant information, offsetting the benefit of stronger supervision. Cal-OPD mitigates this effect by filtering out the TSD-explained component before distillation.

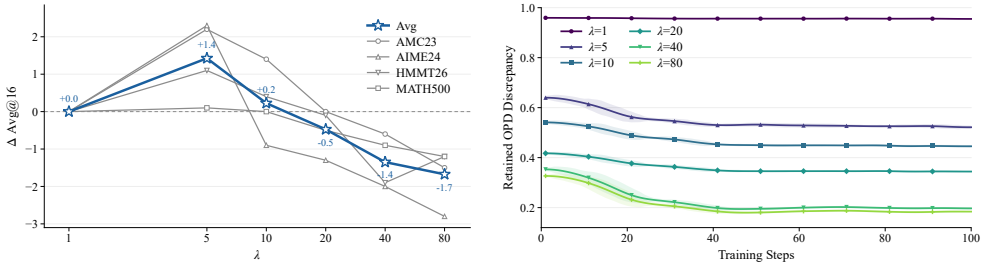
4.3 Analysis and Ablations

Effect of Contextual Interventions on Cal-OPD Performance. Table 4 compares different intervention sets with $\lambda = 5$ throughout, while Figure 4 shows their calibration dynamics. $\mathcal{C}_{\text{eval}}$ achieves the best performance, reaching an average of 53.1. Although evaluative feedback introduces external judgment, it provides no task-specific solution knowledge and yields stronger calibration than instruction interventions, which retain nearly 70% of the teacher–student discrepancy. In contrast, $\mathcal{C}_{\text{sol}}$ causes the largest degradation, with the average falling to 49.0 while retaining only about 20% of the discrepancy. This suggests that solution-induced TSD contains a larger task-relevant component, such that using it for calibration over-filters useful teacher supervision. Overall, $\mathcal{C}_{\text{eval}}$ provides a stronger probe of TSD without the excessive filtering induced by solution-level privilege. Further training dynamics, including entropy and other measures, are provided in Appendix A.7.



(a) Retained OPD Discrepancy **(b) Zero-Advantage Token Ratio** **(c) Average Response Length**

Figure 4: Training dynamics of Cal-OPD under different TSD-estimation interventions for Qwen3-4B-Thinking-2507→Qwen3-1.7B.



(a) Impact of λ on Performance **(b) Impact of λ on Retained OPD Discrepancy**

Figure 5: Effect of the relaxation factor λ on Cal-OPD performance and retained discrepancy ratio.

Effects of Calibration on OPD Training Dynamics. Figure 4 shows the training dynamics of the main $\mathcal{C}_{\text{eval}}$ configuration. The retained-discrepancy ratio decreases from 65% to 52%, while the zero-advantage token ratio increases from 27% to 34%, indicating stronger filtering of the OPD signal. Calibration also alters response-length dynamics. Standard OPD expands the average response length from about 9.8K to 11.8K tokens, whereas Cal-OPD ends at only 9.3K after an initial decrease to 8.5K. Despite the additional teacher computation for estimating TSD, the shorter trajectories make Cal-OPD approximately $1.26\times$ faster to train. Together with the main results, these dynamics show that filtering the TSD-explained component yields a more effective and efficient supervision signal. A complete efficiency analysis is provided in Appendix A.8.

Effect of the Relaxation Factor λ. Figure 5 shows that Cal-OPD achieves optimal performance at $\lambda = 5$, retaining approximately 52% of the teacher–student discrepancy. Increasing λ beyond 5 sharply degrades downstream performance, eventually dropping 1.7 points below the $\lambda = 1$ baseline at $\lambda = 80$. As λ increases, the estimated TSD region expands too broadly, reducing the retained discrepancy to roughly 20% and over-filtering task-relevant supervision. We further compare against a TSD-threshold filtering baseline matched in retained teacher–student discrepancy in Appendix A.9. Cal-OPD remains stronger, indicating that its gains are not due to signal attenuation alone.

5 Conclusion

We show that the teacher–student discrepancy used by standard OPD contains substantial teacher self-deviation (TSD), which emerges even without task-specific knowledge, remains largely insensitive to intervention semantics and correctness, and concentrates on surface-form tokens. Based on these findings, we introduce Calibrated On-Policy Distillation (Cal-OPD), which estimates the teacher's self-deviation region and removes the TSD-explained portion of the OPD signal before optimization. Across different teacher–student scales and mathematical reasoning benchmarks, Cal-OPD consistently outperforms standard OPD and its variants while retaining only 52–65% of the original discrepancy and mitigating response-length expansion during training. These results suggest that effective on-policy distillation depends not on indiscriminately learning more teacher–student discrepancy, but on identifying which discrepancy is truly worth learning.

AI USE STATEMENT

Generative AI tools were used to assist with code generation and implementation during experimentation. All AI-generated code was manually reviewed, verified, and tested by the authors before use. In addition, as described in Appendix A.2, Qwen3-235B-A22B-Instruct-2507 was used to generate candidate reference solutions for data preparation. These solutions were automatically filtered by comparing their final answers with the ground-truth answers, and only correct solutions were retained. The authors take full responsibility for the implementation, experimental results, and final content of this work.

ETHICS STATEMENT

This work studies on-policy distillation for improving the reasoning capabilities of large language models. Our experiments use publicly available models, datasets, and mathematical reasoning benchmarks, and do not involve human subjects, personally identifiable information, or sensitive personal data. The proposed method is intended for general-purpose model training and does not introduce application-specific capabilities that pose additional societal or safety risks beyond those associated with the underlying language models. We follow the applicable licenses and usage terms of all models and datasets used in this work, and our research adheres to the ICLR Code of Ethics. We are not aware of any additional ethical concerns or conflicts of interest arising from this research.

REPRODUCIBILITY STATEMENT

We provide the methodological details and formulation of Cal-OPD in the main text, including the construction of the calibration signal and the resulting training objective. The experimental setup, including the teacher–student configurations, training data, evaluation benchmarks, and key hyperparameters, is described in the experimental section and appendix. Additional implementation details, data construction procedures, ablation studies, and extended analyses are provided in the appendix. To facilitate reproduction, we provide the exact hyperparameters, random seeds, and training scripts in our code release. These materials are intended to provide sufficient information for reproducing the main results and analyses reported in this work.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Hongzhan Chen, Siyue Wu, Xiaojun Quan, Rui Wang, Ming Yan, and Ji Zhang. Mcc-kd: Multi-cot consistent knowledge distillation. In *Findings of the association for computational linguistics: EMNLP 2023*, pp. 6805–6820, 2023.

Chengwei Dai, Kun Li, Wei Zhou, and Songlin Hu. Improve student's reasoning generalizability through cascading decomposed cots distillation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15623–15643, 2024.

Yi Ding and Ruqi Zhang. Does on-policy distillation really distill? from noisy teacher to self-improvement. *arXiv preprint arXiv:2608.31046*, 2026.

Tao Feng, Yicheng Li, Li Chenglin, Hao Chen, Fei Yu, and Yin Zhang. Teaching small language models reasoning through counterfactual distillation. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 5831–5842, 2024.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Qiangqiang He, Zhongheng Wu, and ZiJian Wang. Not all tokens deserve equal credit: Counterfactual sensitivity credit reallocation for long-cot reasoning. *arXiv preprint arXiv:2607.27888*, 2026.

Byeongho Heo, Jaehui Hwang, Sangdoo Yun, and Dongyoon Han. On-policy delta distillation for multilingual math reasoning. *arXiv preprint arXiv:2608.05802*, 2026.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 14852–14882, 2023.

Wenjin Hou, Shangpin Peng, Weinong Wang, Zheng Ruan, Yue Zhang, Zhenglin Zhou, Mingqi Gao, Yifei Chen, Kaiqi Wang, Hongming Yang, et al. Uni-opd: Unifying on-policy distillation with a dual-perspective recipe. *arXiv preprint arXiv:2605.03677*, 2026a.

ZhiYan Hou, Xinyu Tang, Hongyan An, Jianjin Zhang, Weizhen Wang, Yunyun Han, Gengsheng Li, Xiangzhao Hao, Haiyun Guo, Wenbin Hu, et al. Dash: Divergence-adaptive supervision horizons for on-policy self-distillation of reasoning models. *arXiv preprint arXiv:2608.06243*, 2026b.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the association for computational linguistics: ACL 2023*, pp. 8003–8017, 2023.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Yuki Ichihara, Naoto Iwase, Mohammad Atif Quamar, and Junpei Komiyama. Privileged solutions or context-induced teacher behavior? dissecting on-policy self-distillation. *arXiv preprint arXiv:2608.09228*, 2026.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. In *Findings of the association for computational linguistics: EMNLP 2020*, pp. 4163–4174, 2020.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Simran Kaur, Narutatsu Ri, Yinghui He, Liam Fowl, and Sanjeev Arora. Rethinking on-policy self-distillation for thinking models. *arXiv preprint arXiv:2607.05184*, 2026.

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.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

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

Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. Symbolic chain-of-thought distillation: Small models can also "think" step-by-step. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2665–2679, 2023.

Yijiang Li, Bingyang Wang, Yijun Liang, Yunjie Tian, Di Fu, and Nuno Vasconcelos. On-policy self-distillation without any supervision. *arXiv preprint arXiv:2608.06296*, 2026a.

Yuhan Li, Mingxu Zhang, Dazhong Shen, and Ying Sun. Phf: Privileged hidden flow for on-policy self-distillation. *arXiv preprint arXiv:2606.29340*, 2026b.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

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

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*, 2023.

Leyi Pan, Shuchang Tao, Yunpeng Zhai, Lingzhe Zhang, Zhaoyang Liu, Bolin Ding, Aiwei Liu, and Lijie Wen. Rlcsd: Reinforcement learning with contrastive on-policy self-distillation. *arXiv preprint arXiv:2606.11709*, 2026.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. *arXiv preprint arXiv:2602.04942*, 2026.

Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. On-policy self-distillation for reasoning compression. *arXiv e-prints*, pp. arXiv–2603, 2026.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Guobin Shen, Xiang Cheng, Chenxiao Zhao, Lei Huang, Jindong Li, Dongcheng Zhao, and Xing Yu. Anti-self-distillation for reasoning rl via pointwise mutual information. *arXiv preprint arXiv:2605.11609*, 2026a.

Zhanming Shen, Jintao Tong, Shaotian Yan, Chen Shen, Hao Chen, Wentao Ye, Xiaomeng Hu, Rui Miao, Haobo Wang, Junbo Zhao, et al. Purified opsd: On-policy self-distillation without losing how to think. *arXiv preprint arXiv:2607.02234*, 2026b.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Samyak Shrestha and Alexander Tessier. Rethinking privileged information in on-policy self-distillation. *arXiv preprint arXiv:2608.18271*, 2026.

Alex Stein, Furong Huang, and Tom Goldstein. Gates: Self-distillation under privileged context with consensus gating. *arXiv preprint arXiv:2602.20574*, 2026.

Zhiquan Tan and Yinrong Hong. Self-supervised on-policy distillation for reasoning language models. *arXiv preprint arXiv:2605.17497*, 2026.

Kanghui Tian, Siyuan Liu, Ziang Yan, Sheng Xia, Shuai Dong, and Yi Wang. Vicur: Visual cues as recoverable privilege for multimodal on-policy distillation. *arXiv preprint arXiv:2606.05718*, 2026.

Xumeng Wen, Zihan Liu, Shun Zheng, Shengyu Ye, Zhirong Wu, Yang Wang, Zhijian Xu, Xiao Liang, Junjie Li, Ziming Miao, et al. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. In *International Conference on Learning Representations*, volume 2026, pp. 49450–49483, 2026.

Jianyu Wu, Yizhou Wang, Encheng Su, Chen Tang, and Shixiang Tang. Dapd: Dual-anchored policy distillation. *arXiv preprint arXiv:2608.01735*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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*, 2026a.

Yuxiao Yang, Xiaoyun Wang, and Weitong Zhang. Ogls-sd: On-policy self-distillation with outcome-guided logit steering for llm reasoning. *arXiv preprint arXiv:2605.12400*, 2026b.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026a.

Xinlei Yu, Gen Li, Qingyi Si, Guibin Zhang, Yuqi Xu, Congcong Wang, Shuai Dong, Kaiwen Tuo, Xiangyu Zeng, Kaituo Feng, et al. Dopd: Dual on-policy distillation. *arXiv preprint arXiv:2606.30626*, 2026b.

Ke Zhang, Yunjie Tian, Dongdi Zhao, Yijiang Li, Yuanye Liu, Vishal M Patel, and Di Fu. On-policy distillation with best-of-n teacher rollout selection. *arXiv preprint arXiv:2605.09725*, 2026.

Songming Zhang, Xue Zhang, Zengkui Sun, Yufeng Chen, and Jinan Xu. Dual-space knowledge distillation for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 18164–18181, 2024.

Wenhao Zhang. Beyond absolute imitation: Anchored residual guidance for privileged on-policy distillation. *arXiv preprint arXiv:2606.10385*, 2026.

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

Xuyang Zhao, Liting Zhang, Zichen Xu, Zhihu Wang, Xu Caiyue, Shiwan Zhao, and Qicheng Li. Is more privileged information better? from solution traces to problem-solving structure in self-distilled reasoning. *arXiv preprint arXiv:2608.01589*, 2026b.

A Appendix

A.1 Related Work

Knowledge Distillation and On-Policy Distillation. Knowledge distillation (KD) transfers capabilities from a stronger teacher to a weaker student and has become a standard framework for model compression and capability transfer(Hinton et al., 2015). Sequence-level and pretrained language-model distillation extended this paradigm to generation and compression(Kim & Rush, 2016; Sanh et al., 2019; Jiao et al., 2020), while LLM-specific methods introduced reverse or student-aware divergence objectives and output-space alignment(Gu et al., 2024; Ko et al., 2024; Zhang et al., 2024). For reasoning, teacher-generated rationales and reasoning strategies have been transferred through Fine-tune-CoT(Ho et al., 2023), Distilling Step-by-Step(Hsieh et al., 2023), SCoTD(Li et al., 2023), MCC-KD(Chen et al., 2023), Orca 2(Mitra et al., 2023), counterfactual distillation(Feng et al., 2024), and decomposed CoT distillation(Dai et al., 2024); DeepSeek-R1 further demonstrates that CoT reasoning behaviors can be distilled into smaller models(Guo et al., 2025).

Many sequence- and rationale-level distillation methods remain off-policy, training on teacher-generated or otherwise fixed trajectories. On-policy distillation instead provides teacher supervision on trajectories sampled from the student itself, reducing the training–inference distribution mismatch(Agarwal et al., 2024), and has been adopted in large-scale reasoning post-training such as Qwen3(Yang et al., 2025). Recent extensions modify the supervision signal through reward extrapolation(Yang et al., 2026a), entropy- or uncertainty-aware objectives(Jin et al., 2026; Ke et al., 2026), outcome-guided calibration(Hou et al., 2026a), divergence-adaptive supervision horizons(Hou et al., 2026b), and teacher-versus-base delta signals(Heo et al., 2026). On-policy self-distillation further removes the need for a separate stronger teacher through privileged or behavior-conditioned self-teachers(Zhao et al., 2026a; Sang et al., 2026), self-generated supervision(Tan & Hong, 2026), and internal consistency(Li et al., 2026a), while on-policy context distillation internalizes teacher-only context such as prior experience or optimized system prompts(Ye et al., 2026).

Privileged and Context-Augmented Distillation. Recent LLM distillation strengthens the teacher with information unavailable to the student at inference time. On-Policy Self-Distillation (OPSD) conditions a self-teacher on verified reference solutions(Zhao et al., 2026a), SDPO conditions the self-teacher on environment feedback(Hübotter et al., 2026), and privileged-information distillation studies forms of training-only information(Penaloza et al., 2026). Related methods distill teacher-only experience or system prompts(Ye et al., 2026), document-grounded evidence(Stein et al., 2026), or behavioral instructions for reasoning compression(Sang et al., 2026). For mathematical reasoning, Anti-Self-Distillation reverses harmful privileged-teacher guidance(Shen et al., 2026a), DOPD dynamically routes token-level supervision to mitigate privilege illusion(Yu et al., 2026b), PHF transfers privileged hidden-state dynamics(Li et al., 2026b), AR-OPD anchors privileged guidance to a locally compatible view(Zhang, 2026), Purified OPSD removes reference-induced non-transferable components(Shen et al., 2026b), and DAPD introduces dual anchoring against information asymmetry(Wu et al., 2026). Recent analyses further show that privileged context can degrade thinking models(Kaur et al., 2026), that references from other problems can retain comparable gains(Ichihara et al., 2026), that correct references do not provide consistent benefits(Shrestha & Tessier, 2026), and that structured privileged information can outperform raw solution traces(Zhao et al., 2026b). Together, these findings suggest that privileged context changes not only the information available to the teacher, but also the teacher behavior induced by that context, complicating the interpretation of privileged teacher likelihoods as transferable knowledge.

Reliability of Teacher Supervision. Recent work has also questioned the assumption that teacher supervision is uniformly reliable across tokens and trajectories. Entropy-Aware OPD adapts distillation to teacher uncertainty(Jin et al., 2026), Uni-OPD calibrates teacher guidance using outcome-level order consistency(Hou et al., 2026a), and Position-Weighted OPSD shows that teacher-token reliability is strongly structured by reasoning position(Liu et al., 2026). OGLS-SD calibrates privileged teacher logits using outcome contrast(Yang et al., 2026b), while BRTS selects teacher rollouts according to correctness and student alignment(Zhang et al., 2026). Related studies further show that reasoning tokens remain relatively stable while stylistic or surface-form tokens are more sensitive to contextual interventions(Pan et al., 2026; He et al., 2026), and that privileged context can induce shortcut behavior(Tian et al., 2026). More recently, Ding & Zhang (2026) reveal substantial, scale-

Stage	Model	Mode	Max Length	Temp.	Top-p
Solution generation	Qwen3-235B-A22B-Instruct-2507	No-thinking	12,288	1.0	1.0
Student rollout	Qwen3-1.7B	Thinking	16,384	1.0	1.0
Teacher rescoring	Qwen3-8B	Thinking	Fixed rollout	–	–

Table 5: Configurations used for preparing the data for TSD analysis. Solution generation and student rollout collection use one generation per question. Teacher rescoring does not involve sampling, since the student trajectory is kept fixed.

dependent noise in OPD teacher supervision and show that removing noisy supervision can leave student performance largely unchanged. These studies establish that teacher supervision is heterogeneous, but characterize reliability primarily through uncertainty, outcomes, positions, trajectory quality, or specific privileged-reference effects. In contrast, the variation of teacher likelihoods under controlled contextual interventions has not been explicitly modeled as a finite-intervention self-deviation region for calibrating the teacher–student discrepancy in OPD.

A.2 Data Collection and Preparation for TSD Analysis

Our TSD analysis is conducted on a collection of student-generated reasoning trajectories derived from DAPO-17K (Yu et al., 2026a). Since DAPO-17K provides problems and ground-truth answers but does not include reference solutions, we first construct a solution-augmented subset, then generate on-policy student rollouts, and finally rescore the fixed trajectories under different teacher-side contextual interventions. Table 5 summarizes the configurations used in this process.

Reference-Solution Preparation. We use Qwen3-235B-A22B-Instruct-2507 (Yang et al., 2025) to generate one reference solution for each problem in DAPO-17K. The model is used in its no-thinking mode with a maximum response length of 12,288 tokens, temperature 1.0, and top-p 1.0. All generations use the following shared system prompt:

System Prompt

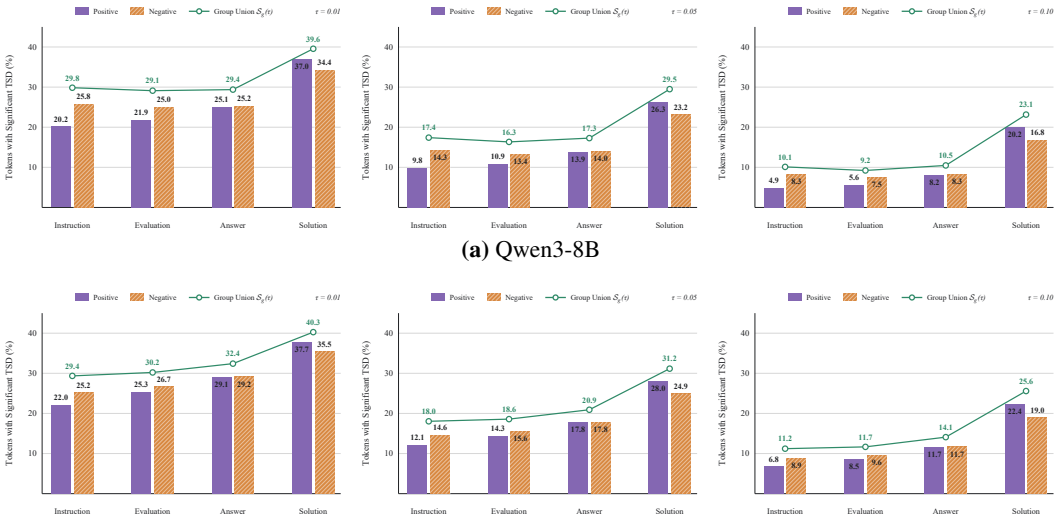
```
You are a helpful math assistant.  Please solve the math problem.
You must enclose your final answer exactly within \boxed{}.
```

We extract the final answer enclosed in `\boxed{}` and compare it with the ground-truth answer provided by DAPO-17K. Solutions with incorrect final answers are discarded. After filtering, we retain 15,560 question–solution pairs, covering 90.18% of the 17,255 problems, with 1,695 problems unsolved. The retained reference solutions contain an average of 7,636 characters, a median of 5,679 characters, a 95th percentile of 19,493 characters, and a range of 445–39,695 characters.

Student Rollout Collection. Starting from the 15,560 retained questions, we generate one on-policy response per question using Qwen3-1.7B (Yang et al., 2025) in thinking mode. We set the maximum response length to 16,384 tokens, temperature to 1.0, and top-p to 1.0, while using the same system prompt as above. Rollout generation is stopped once the cumulative number of student-generated response tokens reaches approximately 60 million. This results in 6,528 question–response pairs used for subsequent TSD analysis. The student responses contain an average of 9,270 tokens and a median of 7,758 tokens.

Training Data Usage. All distillation methods in our main experiments are trained on the same 15,560 question–solution pairs.

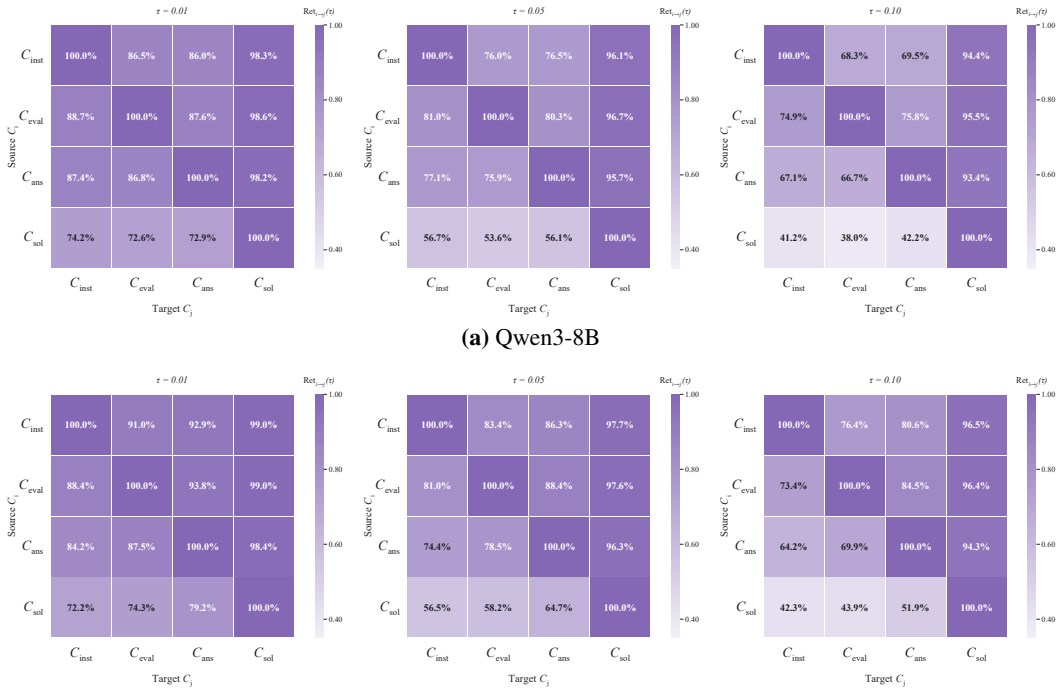
Construction of Privileged Contexts. For the answer-level interventions in Table 1, the positive variant uses the ground-truth answer associated with the current problem. To construct $c_{\text{ans}}^{\text{neg}}$, we randomly generate an incorrect answer with the same number of digits as the corresponding correct answer, thereby controlling for differences in answer length.



(a) Qwen3-8B

(b) Qwen3-4B-Thinking-2507

Figure 6: Prevalence of significant TSD across thresholds and teacher models. Although the absolute prevalence decreases with stricter thresholds, substantial TSD continues to emerge under task-agnostic interventions, while solution-level privilege consistently broadens the affected set.

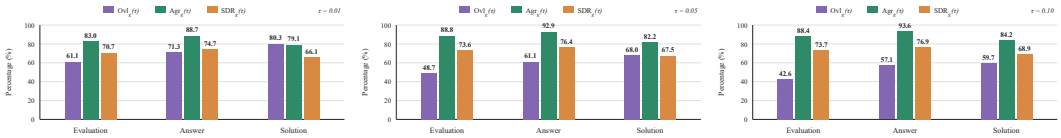


(a) Qwen3-8B

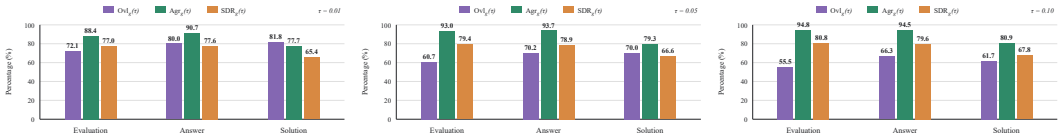
(b) Qwen3-4B-Thinking-2507

Figure 7: Retention of significant TSD across thresholds and teacher models. Less informative interventions retain nearly all affected token positions under solution-level privilege, whereas retention in the reverse direction is substantially lower.

For the solution-level interventions, $c_{\mathrm{sol}}^{\mathrm{pos}}$ uses the verified reference solution generated for the current problem. To construct $c_{\mathrm{sol}}^{\mathrm{neg}}$, we randomly select the reference solution of a different problem whose tokenized length differs from that of the current reference solution by less than 10%. This length-



(a) Qwen3-8B



(b) Qwen3-4B-Thinking-2507

Figure 8: Semantic consistency of TSD across thresholds and teacher models. Contrasting interventions consistently exhibit substantial positional overlap, high directional agreement, and predominantly shared deviation across thresholds and model scales.

matching constraint reduces the possibility that differences between positive and negative solution-level interventions are driven primarily by contextual length rather than content.

Teacher Likelihood Rescoring. We use Qwen3-8B (Yang et al., 2025) as the teacher and evaluate each fixed Qwen3-1.7B rollout under nine teacher contexts:

$$c_0, \quad c_{\text{inst}}^{\text{pos}}, \; c_{\text{inst}}^{\text{neg}}, \; c_{\text{eval}}^{\text{pos}}, \; c_{\text{eval}}^{\text{neg}}, \; c_{\text{ans}}^{\text{pos}}, \; c_{\text{ans}}^{\text{neg}}, \; c_{\text{sol}}^{\text{pos}}, \; c_{\text{sol}}^{\text{neg}}. \tag{15}$$

The shared system prompt above is included in every condition and is therefore not itself treated as a contextual intervention. For all nine contexts, the problem x, student rollout y, prefix $y_{<t}$, and evaluated token y_t are kept identical; only the additional teacher-side context is changed. The teacher does not generate new trajectories. Instead, we recompute the token-level log-likelihood assigned to every student-generated token under each context, allowing the resulting likelihood differences to be attributed directly to the contextual intervention.

Consequently, the TSD analysis is based on approximately 60 million student-generated tokens, with each token rescored under the baseline context and eight contextual interventions.

A.3 TSD Semantic Consistency Across Thresholds and Model Scales

We further examine whether the TSD patterns reported in the main text persist across significance thresholds and teacher scales. Using the same Qwen3-1.7B student trajectories described in Appendix A.2, we repeat the analysis with Qwen3-8B and Qwen3-4B-Thinking-2507 as teachers under $\tau \in \{0.01, 0.05, 0.10\}$. We consider the prevalence of significant TSD, retention across intervention groups, and semantic consistency across the six teacher–threshold configurations.

Prevalence of Significant TSD. Figure 6 shows that increasing τ reduces the absolute prevalence of significant TSD, while preserving the relative pattern across intervention groups. For Qwen3-8B, the union prevalence under task-agnostic instructions decreases from 29.8% at $\tau = 0.01$ to 10.1% at $\tau = 0.10$, yet remains comparable to evaluative feedback at 29.1% and 9.2%, respectively. The same pattern holds for Qwen3-4B-Thinking-2507, where task-agnostic instructions remain close to evaluative feedback across all thresholds. Solution-level privilege consistently produces the highest prevalence, supporting the conclusion that richer privileged context broadens the affected set rather than being necessary for TSD to emerge.

Retention Across Intervention Groups. Figure 7 exhibits a consistent asymmetric retention pattern across thresholds and teacher models. Token positions affected under instruction, evaluation, and answer interventions are largely retained under solution-level privilege. Even at $\tau = 0.10$, their retention into the solution group remains at least 93.4% for Qwen3-8B and 94.3% for Qwen3-4B-Thinking-2507. The reverse retention is substantially lower and decreases as the threshold becomes stricter. This asymmetry indicates that richer privileged context predominantly extends an existing set of TSD-sensitive positions rather than replacing it with a distinct deviation pattern.

Table 6: Top-24 and bottom-24 token forms under $c_{\text{inst}}^{\text{pos}}$ and $c_{\text{inst}}^{\text{neg}}$ at $\tau = 0.01$. The final row reports unweighted means of ρ_τ and $\overline{|\Delta|}$ over each 24-token subset.

#	$c_{\text{inst}}^{\text{pos}}$						$c_{\text{inst}}^{\text{neg}}$					
	Highest ρ_τ			Lowest ρ_τ			Highest ρ_τ			Lowest ρ_τ		
	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$
1	maybe	74.4	0.066		1.0	0.069	maybe	84.4	0.110		1.5	0.080
2	consider	73.1	0.071	###	2.1	0.071	consider	84.2	0.106	0	2.9	0.116
3	try	66.1	0.086	0	2.2	0.095	however	82.8	0.157	###	3.4	0.082
4	however	65.2	0.086	°	2.6	0.068	therefore	82.1	0.170	°	3.7	0.083
5	since	64.4	0.080	√	2.9	0.084	earlier	80.9	0.121	_	3.7	0.089
6	if	63.9	0.072	_	3.0	0.072	try	80.9	0.177	9	3.9	0.119
7	earlier	63.2	0.066	9	3.1	0.099	now	78.5	0.165	√	4.0	0.120
8	now	63.0	0.094	θ	3.2	0.076	how	75.1	0.112	8	4.1	0.124
9	how	62.8	0.078	8	3.3	0.102	because	74.6	0.114	frac	4.2	0.073
10	think	61.8	0.070	frac	3.3	0.062	another	74.4	0.127	6	4.3	0.125
11	another	60.8	0.088	6	3.4	0.101	if	74.2	0.102	θ	4.4	0.105
12	check	60.6	0.075	7	3.5	0.101	since	74.1	0.109	7	4.4	0.124
13	because	60.2	0.080	5	3.6	0.096	think	73.9	0.108	5	4.5	0.119
14	therefore	58.9	0.088	{	3.6	0.067	let	73.6	0.122	_i	4.6	0.090
15	when	58.9	0.078	2	3.8	0.089	so	72.7	0.118	{	4.7	0.088
16	there	58.7	0.073	_i	3.8	0.077	but	72.0	0.113	2	4.8	0.110
17	let	58.0	0.076	)^	4.0	0.078	this	71.8	0.117	4	5.1	0.119
18	given	57.2	0.105	+	4.0	0.089	altern.	71.7	0.123	1	5.3	0.109
19	also	56.8	0.083	4	4.0	0.095	here	71.2	0.115	)^	5.4	0.091
20	use	56.1	0.081	1	4.2	0.088	when	71.1	0.110	3	5.5	0.112
21	so	55.8	0.081	3	4.2	0.092	there	69.2	0.099	+	5.6	0.106
22	but	55.7	0.077	't	4.6	0.088	previous	69.2	0.086	't	5.7	0.101
23	we	55.2	0.065	)/	5.1	0.077	it	69.1	0.117	cdot	7.2	0.057
24	yes	55.1	0.096	cdot	5.7	0.051	seems	68.8	0.097	)/	7.4	0.100
Mean–		61.1	0.080	–	3.5	0.083	–	75.0	0.121	–	4.6	0.102

Semantic Consistency. Figure 8 shows that the semantic-consistency pattern also persists across thresholds and teacher models, and across all three intervention groups. As τ increases, positional overlap generally decreases because fewer tokens remain significant under both contrasting interventions, while directional agreement remains consistently high. At $\tau = 0.10$, answer-level agreement reaches 93.6% for Qwen3-8B and 94.5% for Qwen3-4B-Thinking-2507, with corresponding SDR values of 76.9% and 79.6%. Evaluative interventions show the same trend, whereas solution-level interventions consistently exhibit higher positional overlap but lower SDR. Thus, stricter thresholds change which tokens remain significant without altering the broader finding that contrasting semantics induce strongly aligned and predominantly shared TSD.

Across all configurations, the qualitative conclusions of the main analysis remain unchanged. TSD emerges substantially without task-specific knowledge, richer privileged context largely preserves and expands previously affected token positions, and contrasting intervention semantics leave much of the resulting TSD directionally aligned and shared. **These results show that the observed TSD patterns are not specific to a particular significance threshold or teacher scale.**

A.4 Token-Level TSD Patterns Across Contextual Interventions

To examine whether the token-level structure of TSD depends on a particular contextual intervention, we repeat the token-form analysis under all eight contextual interventions using Qwen3-8B at $\tau = 0.01$. Following the main analysis, we merge tokenizer variants corresponding to the same normalized surface form and retain forms occurring more than 20,000 times. Tables 6–9 report the top-24 and bottom-24 forms ranked by $\rho_\tau(v)$ for each positive–negative intervention pair, together with their significant-TSD rates and mean deviation magnitudes.

Stability Across Intervention Polarity. Positive and negative variants within each intervention group produce substantially overlapping token rankings. Their top-24 sets share $18/24$ forms for instruction, $22/24$ for evaluation, $24/24$ for answer-level privilege, and $23/24$ for solution-level privilege. The corresponding bottom-24 overlaps are $24/24$, $23/24$, $24/24$, and $24/24$. Thus, reversing intervention polarity or correctness generally leaves the token forms most prone to TSD

Table 7: Top-24 and bottom-24 token forms under $c_{\text{eval}}^{\text{pos}}$ and $c_{\text{eval}}^{\text{neg}}$ at $\tau = 0.01$. The final row reports unweighted means of ρ_τ and $\overline{|\Delta|}$ over each 24-token subset.

#	$c_{\text{eval}}^{\text{pos}}$						$c_{\text{eval}}^{\text{neg}}$					
	Highest ρ_τ			Lowest ρ_τ			Highest ρ_τ			Lowest ρ_τ		
	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$
1	maybe	77.5	0.079	}{	1.0	0.072	maybe	83.8	0.100	}{	1.1	0.076
2	consider	76.2	0.074	###	1.1	0.047	however	83.2	0.139	###	1.3	0.053
3	earlier	72.4	0.082	0	2.5	0.111	consider	82.6	0.091	0	2.9	0.124
4	however	72.2	0.102	frac	2.8	0.058	earlier	79.9	0.113	frac	3.1	0.064
5	try	69.2	0.091	°	2.9	0.070	therefore	77.9	0.131	_	3.6	0.092
6	therefore	67.4	0.098	√	3.2	0.093	try	77.1	0.134	°	3.7	0.068
7	now	66.5	0.102	_	3.3	0.076	says	74.6	0.090	9	3.9	0.138
8	since	66.2	0.083	9	3.4	0.114	another	74.5	0.118	√	3.9	0.108
9	how	66.1	0.081	θ	3.4	0.082	since	74.3	0.103	θ	4.1	0.099
10	if	66.1	0.077	8	3.6	0.119	think	73.4	0.093	8	4.1	0.140
11	think	65.4	0.074	6	3.7	0.118	if	73.4	0.090	6	4.3	0.138
12	because	65.1	0.086	{	3.8	0.074	how	72.9	0.104	_i	4.4	0.087
13	another	64.5	0.091	7	3.9	0.114	altern.	72.8	0.120	{	4.4	0.078
14	previous	63.6	0.071	5	3.9	0.111	previous	72.6	0.087	7	4.5	0.138
15	says	62.8	0.073	_i	4.0	0.084	now	72.5	0.149	5	4.5	0.130
16	when	62.7	0.084	2	4.2	0.103	because	72.0	0.105	cdot	4.6	0.053
17	there	62.2	0.078	cdot	4.4	0.047	let	71.0	0.133	2	4.9	0.112
18	so	62.2	0.087	4	4.5	0.111	but	70.8	0.100	4	5.0	0.127
19	but	61.9	0.082	)^	4.5	0.087	here	70.0	0.103	)^	5.2	0.091
20	let	61.9	0.084	+	4.5	0.099	check	69.9	0.098	1	5.3	0.112
21	this	61.1	0.086	1	4.6	0.103	seems	69.9	0.090	3	5.4	0.130
22	seems	60.7	0.073	3	4.8	0.104	so	69.8	0.106	't	5.5	0.124
23	also	60.3	0.087	't	5.3	0.091	when	69.3	0.100	+	5.6	0.106
24	altern.	60.2	0.085	)/	5.8	0.087	there	68.9	0.095	}	6.7	0.069
Mean	–	65.6	0.084	–	3.7	0.091	–	74.0	0.108	–	4.3	0.102

Table 8: Top-24 and bottom-24 token forms under $c_{\text{ans}}^{\text{pos}}$ and $c_{\text{ans}}^{\text{neg}}$ at $\tau = 0.01$. The final row reports unweighted means of ρ_τ and $\overline{|\Delta|}$ over each 24-token subset.

#	$c_{\text{ans}}^{\text{pos}}$						$c_{\text{ans}}^{\text{neg}}$					
	Highest ρ_τ			Lowest ρ_τ			Highest ρ_τ			Lowest ρ_τ		
	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$
1	maybe	83.3	0.254	}{	1.1	0.070	maybe	83.7	0.362	}{	1.1	0.071
2	however	82.2	0.210	###	1.6	0.058	however	82.3	0.163	###	1.6	0.055
3	therefore	80.6	0.210	0	2.8	0.143	consider	81.0	0.090	0	2.8	0.131
4	consider	80.1	0.090	frac	3.1	0.066	therefore	80.6	0.358	frac	3.1	0.062
5	earlier	79.5	0.214	_	3.7	0.088	earlier	79.5	0.226	_	3.6	0.095
6	if	74.7	0.135	√	3.7	0.120	if	75.9	0.176	√	3.7	0.119
7	try	74.6	0.121	θ	3.8	0.088	another	75.6	0.431	9	3.9	0.146
8	seems	74.6	0.458	9	3.9	0.155	try	75.1	0.126	θ	3.9	0.086
9	another	73.9	0.211	8	4.2	0.165	says	74.4	0.111	8	4.1	0.152
10	because	73.2	0.127	°	4.2	0.093	seems	74.0	0.487	°	4.2	0.087
11	now	73.1	0.154	6	4.3	0.157	how	73.5	0.170	6	4.3	0.145
12	how	73.1	0.158	_i	4.4	0.082	think	73.4	0.189	_i	4.3	0.085
13	says	73.0	0.106	{	4.4	0.086	because	73.4	0.136	{	4.5	0.085
14	think	72.6	0.126	5	4.5	0.152	since	73.2	0.136	5	4.5	0.139
15	since	72.1	0.125	7	4.5	0.161	check	72.4	0.154	7	4.5	0.150
16	but	72.0	0.188	2	4.8	0.129	there	72.2	0.200	2	4.8	0.120
17	so	71.4	0.142	)^	5.0	0.087	now	72.1	0.151	4	5.0	0.136
18	check	71.1	0.122	4	5.1	0.141	but	71.8	0.136	)^	5.1	0.087
19	let	70.7	0.426	cdot	5.2	0.049	so	71.4	0.183	cdot	5.3	0.049
20	there	70.3	0.141	1	5.3	0.133	altern.	71.2	0.346	1	5.3	0.127
21	altern.	70.2	0.181	3	5.4	0.137	let	71.2	1.088	3	5.4	0.132
22	previous	70.1	0.125	+	5.7	0.105	previous	70.0	0.113	+	5.8	0.104
23	when	69.2	0.123	't	5.9	0.120	when	69.9	0.150	't	5.8	0.120
24	wait	68.1	0.221	}	7.0	0.090	wait	68.7	0.175	}	7.3	0.112
Mean	–	73.9	0.182	–	4.3	0.111	–	74.4	0.244	–	4.3	0.108

largely unchanged. This is especially true for forms that remain comparatively stable across conditions, while the instruction-level top set shows somewhat more variation than the other pairs.

Table 9: Top-24 and bottom-24 token forms under $c_{\text{sol}}^{\text{pos}}$ and $c_{\text{sol}}^{\text{neg}}$ at $\tau = 0.01$. The final row reports unweighted means of ρ_τ and $\overline{|\Delta|}$ over each 24-token subset.

#	$c_{\text{sol}}^{\text{pos}}$						$c_{\text{sol}}^{\text{neg}}$					
	Highest ρ_τ			**Lowest ρ_τ**			**Highest ρ_τ**			**Lowest ρ_τ**		
	Token	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	**Token**	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	**Token**	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$	**Token**	ρ_τ (%)	$\overline{\lvert\Delta\rvert}$
1	maybe	97.4	0.594	}{	2.8	0.251	maybe	96.2	0.305	}{	2.0	0.141
2	however	97.4	0.625	###	3.2	0.131	however	95.3	0.358	###	2.9	0.102
3	therefore	95.9	0.460	0	5.0	0.356	therefore	95.0	0.351	0	4.3	0.250
4	consider	95.4	0.380	9	6.8	0.384	consider	93.5	0.230	9	5.9	0.270
5	altern.	94.2	0.576	8	7.2	0.395	earlier	92.5	0.304	frac	6.0	0.110
6	earlier	94.2	0.413	√	7.2	0.278	altern.	90.3	0.309	_	6.0	0.172
7	seems	93.6	0.408	°	7.3	0.170	try	90.2	0.281	√	6.2	0.205
8	another	92.9	0.504	_	7.3	0.287	another	90.1	0.287	8	6.2	0.271
9	since	91.8	0.441	6	7.4	0.398	seems	90.0	0.229	°	6.2	0.117
10	try	91.8	0.404	7	7.7	0.389	think	88.6	0.244	6	6.4	0.278
11	think	91.6	0.402	5	7.8	0.379	says	88.6	0.213	7	6.8	0.266
12	says	90.4	0.246	θ	7.9	0.280	since	88.4	0.247	{	6.8	0.153
13	wait	90.0	0.415	frac	8.0	0.225	here	88.0	0.266	5	6.8	0.257
14	check	89.6	0.278	{	8.0	0.234	if	86.9	0.242	θ	6.9	0.215
15	let	89.5	0.504	4	8.8	0.372	because	86.7	0.246	_i	7.6	0.181
16	here	89.4	0.364	2	8.8	0.323	wait	86.7	0.226	4	7.6	0.256
17	because	89.4	0.396	3	9.2	0.362	previous	86.3	0.219	2	7.7	0.219
18	now	89.2	0.485	_i	9.2	0.283	how	86.2	0.264	3	8.1	0.259
19	also	89.2	0.470	1	9.4	0.330	so	86.2	0.263	1	8.2	0.218
20	if	89.2	0.364	't	10.0	0.302	check	86.0	0.210	't	8.8	0.238
21	similarly	89.0	0.532	)^	10.5	0.208	also	85.8	0.271	)^	9.0	0.166
22	but	88.7	0.439	+	10.7	0.276	but	85.7	0.221	cdot	9.4	0.094
23	so	88.3	0.366	cdot	12.8	0.159	now	85.5	0.310	+	9.5	0.203
24	how	88.3	0.408	}	12.9	0.188	let	84.8	0.267	}	11.1	0.130
Mean	–	91.5	0.436	–	8.2	0.290	–	88.9	0.265	–	6.9	0.199

Stability Across Intervention Types. The same structure persists across different types of contextual intervention. Any two of the eight intervention conditions share at least $18/24$ top-ranked forms and $22/24$ bottom-ranked forms, while $16/24$ top forms and $22/24$ bottom forms appear in every ranking. Across all conditions, high-ρ_τ forms are consistently dominated by natural-language surface expressions such as `maybe`, `however`, `therefore`, `consider`, and `think`, whereas low-ρ_τ forms are dominated by digits, mathematical symbols, and notation. This separation is also quantitatively strong: the mean top-24 ρ_τ ranges from 61.1% to 91.5%, compared with only 3.5% to 8.2% for the bottom-24, corresponding to an approximately 11–18$\times$ gap within individual conditions.

Context Modulates Strength More Than Token Identity. While the ranking structure remains stable, the magnitude of TSD varies substantially across contextual interventions. Under positive interventions, the mean $\overline{|\Delta|}$ of the top-24 forms increases from 0.080 under instruction and 0.084 under evaluation to 0.182 under answer-level privilege and 0.436 under solution-level privilege. The bottom-24 exhibits the same overall trend, rising from 0.083 and 0.091 to 0.111 and 0.290, respectively. Negative interventions show a similar overall increase: the top-24 means are 0.121, 0.108, 0.244, and 0.265 for instruction, evaluation, answer-level privilege, and solution-level privilege, respectively, while the corresponding bottom-24 means are 0.102, 0.102, 0.108, and 0.199. Thus, richer privileged context can substantially amplify teacher-side likelihood shifts, particularly at the solution level. Crucially, however, this amplification does not substantially alter which token forms occupy the two ends of the ranking: largely the same surface-form tokens remain TSD-prone, while numerical and symbolic forms remain comparatively stable. These results indicate that contextual interventions primarily modulate the strength of teacher self-deviation, whereas its token-level concentration remains stable across intervention semantics and privilege types.

A.5 Trajectory-Level Case Study of TSD

We complement the aggregate analysis with two trajectory-level examples illustrating how TSD is distributed within complete reasoning traces. We use the same Qwen3-1.7B student and Qwen3-8B teacher configuration and visualize TSD under $c_{\text{sol}}^{\text{pos}}$. Red and blue indicate decreases and increases

in teacher log-likelihood, respectively, with darker shades denoting larger $|\Delta_t^T|$; tokens with $|\Delta_t^T| \leq 0.1$ are left uncolored. Since the student trajectory is fixed, all highlighted variation is induced solely by the additional reference solution.

Case 1: Correct Student Rollout *Outcome: Correct*

Problem. Calculate the least integer greater than $5^{(-6)(-5)(-4)\cdots(2)(3)(4)}$.

Answer. The student rollout outputs $\boxed{2}$, which matches the gold answer.

TSD Legend. $[-0.3, -0.1)$ $[-0.7, -0.3)$ $[-1.5, -0.7)$ $[-3, -1.5)$ < -3
$(0.1, 0.3]$ $(0.3, 0.7]$ $(0.7, 1.5]$ $(1.5, 3]$ > 3

Student trajectory excerpt under $c_{\text{sol}}^{\text{pos}}$.

Okay, so I need to find the least integer greater than $5^-(-6)(-5)(-4)...(2)(3)(4)^$. Hmm, let me start by understanding the exponent here. The exponent is the product of all integers from -6 up to 4. Wait, let me confirm: the exponent is (-6)(-5)(-4)...(2)(3)(4). So that's multiplying all integers from -6 to 4, but I need to check if that's correct. Let me count how many numbers are there. Starting from -6 up to 4. Let's see: from -6 to 4 inclusive. The number of terms would be 4 - (-6) + 1 = 11 terms. But wait, let me check: Numbers: -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4. Yes, that's 11 numbers. But wait, the problem says "(-6)(-5)(-4)...(2)(3)(4)". So the product is from -6 to 4, including 0? Wait, but 0 is included here. However, when you multiply by 0, the entire product becomes 0. But then the exponent would be 0, so 5^0 is 1. Then the least integer greater than 1 is 2. But that seems too straightforward. Maybe I made a mistake here. Wait, but hold on. Let me check again. The problem says "(-6)(-5)(-4)...(2)(3)(4)". Is the product from -6 up to 4, including 0? But if that's the case, then the product is zero, so the exponent is zero, and 5^0 is 1. Therefore, the least integer greater than 1 is 2. But that seems too easy. Maybe there's a misunderstanding here. Alternatively, maybe the product is from -6 to 4, excluding 0? But the problem didn't specify. Wait, the original problem says "(-6)(-5)(-4)...(2)(3)(4)". The ellipsis is between -4 and 2, so maybe the product is from -6 up to 4, but skipping 0? Wait, but how? If you have -6, -5, -4, ..., 2, 3, 4. So if you start at -6 and go up to 4, but not including 0? But 0 is between -6 and 4. So unless there's a typo, but the problem is written as (-6)(-5)(-4)...(2)(3)(4). So maybe the product is from -6 to 4, including 0? But then the product is zero. But maybe there's a mistake here. Wait, but let me check if the product is from -6 to 4, but not including 0. If that's the case, then the product is (-6)(-5)(-4)...(-1)(1)(2)(3)(4). Let's check how many terms that would be. From -6 to -1 is 6 numbers, then 1, 2, 3, 4. So total of 6 + 4 = 10 terms. But then the product would be (-6)(-5)...(-1) * 1 * 2 * 3 * 4. Let me compute that.

First, the product from -6 to -1 is (-1)^6 * 6! = 1 * 720 = 720. Then multiply by 1*2*3*4 = 24. So total product is 720 * 24 = 17,280. Then 5^17280 is an extremely large number. Then the least integer greater than that would be 5^17280 + 1, but that seems way too big. But the problem says "the least integer greater than 5^-...^" so maybe the answer is something else. But the original problem is written as (-6)(-5)...(2)(3)(4). So if the product includes 0, then it's zero, so 5^0 is 1. Then the least integer greater than 1 is 2. But maybe the problem is written with a different intention.

Wait, maybe the exponent is from -6 to 4, but not including 0? But how? If you have from -6 to 4, but not including 0, then the product is (-6)(-5)...(-1)(1)(2)(3)(4). Which is what I thought earlier. But then the exponent is 17280, so 5^17280 is a gigantic number, and the least integer greater than that would be 5^17280 + 1. But that seems like an unreasonable answer. However, maybe there's a mistake in my calculation. Wait, let me check the product again. If the product is from -6 to 4, excluding 0, then:

First, the product from -6 to -1 is (-6)(-5)(-4)(-3)(-2)(-1). Let me compute that:

(-6) * (-5) = 30
30 * (-4) = -120
-120 * (-3) = 360
360 * (-2) = -720
-720 * (-1) = 720. So that's 720.

[Remaining trajectory omitted.]

Case 2: Incorrect Student Rollout *Outcome: Incorrect*

Problem. Alex the Kat has written 61 problems for a math contest, and there are a total of 187 problems submitted. How many more problems does he need to write and submit before he has written half of the total problems?

Answer and error. The student outputs 93.5, whereas the gold answer is 65. It incorrectly keeps the total fixed at 187 and treats $187/2 = 93.5$ as the required additional count.

TSD Legend. $[-0.3, -0.1)$ $[-0.7, -0.3)$ $[-1.5, -0.7)$ $[-3, -1.5)$ < -3
$(0.1, 0.3]$ $(0.3, 0.7]$ $(0.7, 1.5]$ $(1.5, 3]$ > 3

Student trajectory excerpt under $c_{\text{sol}}^{\text{pos}}$.

Okay, let's try to figure out this problem step by step. So, Alex the Kat has already written 61 problems for a math contest. The total number of problems submitted is 187. The question is asking how many more problems he needs to write and submit before he has written half of the total problems.
First, I need to understand what exactly is being asked here. The total number of problems submitted is 187, but I think that might not be directly relevant. Wait, actually, the problem says Alex has written 61 problems, and there are a total of 187 problems submitted. Hmm, maybe I need to clarify that.
Wait, maybe the total number of problems in the contest is 187? Or is 187 the number of problems submitted by others? The wording is a bit confusing. Let me read it again: "Alex the Kat has written 61 problems for a math contest, and there are a total of 187 problems submitted. How many more problems does he need to write and submit before he has written half of the total problems?"
So, the total number of problems submitted is 187. But Alex has written 61. Wait, maybe "submitted" refers to the total number of problems that have been submitted by everyone, including Alex? Or is 187 the total number of problems in the contest? This is a bit ambiguous. But let's think. If Alex has written 61 problems, and there are 187 problems submitted in total, maybe the total number of problems in the contest is 187? But then he has written 61, so he needs to write half of 187, which is 93.5? But that doesn't make sense because you can't write half a problem. Wait, maybe the total number of problems is different.
Alternatively, maybe the total number of problems in the contest is 187, and Alex has written 61. Then he needs to write half of the total, which is $187/2 = 93.5$. But since you can't have half a problem, maybe it's 94? But the problem says "how many more problems does he need to write and submit," so maybe it's expecting a fractional answer? But that seems odd.
Wait, maybe the total number of problems is not 187. Let me re-examine the problem. "Alex the Kat has written 61 problems for a math contest, and there are a total of 187 problems submitted. How many more problems does he need to write and submit before he has written half of the total problems?"
Hmm. Maybe "submitted" refers to the total number of problems that have been submitted by all participants, including Alex? So, Alex has written 61 problems, and there are 187 problems submitted in total. So, the total number of problems in the contest is 187? But then Alex has written 61, so he needs to write half of 187, which is 93.5. But since you can't write half a problem, maybe the answer is 94? But the problem is probably expecting an exact answer.
Alternatively, maybe the total number of problems in the contest is not 187. Maybe 187 is the total number of problems that have been submitted, and Alex has written 61. Then, the total number of problems in the contest is 187? But then he needs to write half of that, which is 93.5. But again, this seems odd.
Wait, maybe the problem is that Alex has written 61 problems, and there are 187 problems in total. So, he needs to write half of 187, which is 93.5. But since he can't write half a problem, maybe we need to round up? But the problem might be expecting an exact answer.
Alternatively, maybe the problem is that the total number of problems submitted is 187, and Alex has written 61. So, the total number of problems that have been submitted is 187. But Alex needs to write half of the total problems. Wait, but if the total problems submitted is 187, then half of that is 93.5. So, he needs to write 93.5 more problems? But that's not possible.
Wait, maybe I need to think differently. Maybe the total number of problems in the contest is 187. Alex has written 61, so he needs to write half of 187, which is 93.5. Therefore, he needs to write 93.5 more problems. But since you can't write half a problem, maybe the answer is 94? But the problem says "how many more problems does he need to write and submit," so maybe it's expecting a fractional answer? But in reality, problems are whole numbers.

[Remaining trajectory omitted.]

TSD Throughout a Correct Trajectory. Case 1 shows a rollout that correctly concludes $\boxed{2}$ after recognizing that the exponent contains zero. Despite reaching the correct reasoning early, the student repeatedly revisits its interpretation. TSD is distributed throughout this process rather than being confined to task-critical mathematical steps. Strong deviations frequently occur on discourse and reasoning-management expressions such as `Hmm`, `Wait`, `But`, `However`, `Maybe`, and `Alternatively`, whereas much of the numerical and symbolic content remains comparatively stable. This qualitatively matches the token-level pattern in Table 2.

TSD Is Not Necessarily Localized Around Reasoning Errors. Case 2 shows an incorrect rollout in which the student repeatedly treats 187 as a fixed total and reasons from $187/2 = 93.5$. The correct relation is

$$61 + x = \frac{187 + x}{2}, \tag{16}$$

where x denotes the number of additional problems Alex needs to write, which yields $x = 65$. However, the TSD pattern does not concentrate around this identifiable conceptual error or the repeated occurrence of 93.5. Instead, substantial deviations again appear broadly on expressions such as `First`, `Hmm`, `Wait`, `Alternatively`, and `Therefore`. Thus, even when the privileged solution directly resolves the student's mistake, the induced likelihood variation does not behave as a localized reasoning-error signal.

Takeaway. Across both correct and incorrect trajectories, TSD is broadly distributed and particularly pronounced on tokens that organize or redirect reasoning, rather than selectively aligning with mathematical correctness or error locations. These examples provide a trajectory-level illustration of why context-induced teacher variation should not be indiscriminately treated as transferable supervision, further motivating the calibration mechanism in Cal-OPD.

A.6 Training Details and Hyperparameters

We implement all methods in the `verl` framework (Sheng et al., 2025) and train on 8 NVIDIA H20 GPUs, with 4 GPUs hosting the student and 4 hosting the teacher. All methods share a common training recipe unless otherwise specified. We first summarize the shared configuration, then describe method-specific hyperparameters and the algorithmic role each plays.

Shared Training Configuration. Table 10 lists the optimization and training-loop settings common to all baselines and Cal-OPD. All methods are trained for 100 steps, with a maximum response length of 16,384 tokens during training, temperature 1.0, and top-p 1.0. We use AdamW with a learning rate of 1×10^{-6}, no learning-rate warmup, weight decay 0.01, a cosine schedule, and gradient clipping at 1.0. Each training step processes 256 on-policy trajectories, with one rollout per question, one PPO epoch, and a mini-batch size of 256. The entropy coefficient and KL regularization are set to zero because the OPD distillation objective replaces the standard RL KL penalty. We use GRPO as the advantage estimator with standard deviation normalization within each group (Yu et al., 2026a).

Distillation-Specific Configuration. The distillation module operates on student-generated trajectories and queries the teacher for token-level log-probabilities. We set the student chunk size to 1024 and the teacher chunk size to 128, which controls the number of tokens processed per forward pass and is chosen to balance GPU memory against throughput. The teacher returns log-probabilities over its top-16 tokens by default, providing a dense yet bounded supervision signal. All student-generated tokens participate in the distillation loss without filtering, corresponding to a token selection ratio of 1.0 with random selection. The policy loss is computed in `reinforce` mode, and the per-token loss is clamped at a maximum value of 10 to prevent destabilizing updates from outlier advantages.

Method-Specific Hyperparameters. Table 11 summarizes the hyperparameters that differ across baselines. Below, we describe the algorithmic role of each.

ExOPD. ExOPD extrapolates the teacher-derived reward beyond the teacher's own performance level with a scaling factor $\lambda_{\text{exo}} > 1$ (Yang et al., 2026a); we set $\lambda_{\text{exo}} = 1.25$. It maintains

Table 10: Shared training configuration for all methods.

Category	Hyperparameter	Value
Optimizer		
	Learning rate	1×10^{-6}
	LR warmup ratio	0.0
	Weight decay	0.01
	LR scheduler	cosine
	Gradient clip	1.0
Training loop		
	Training steps	100
	Max prompt length	2,048
	Max response length	16,384
	Temperature	1.0
	Top-p	1.0
	PPO epochs	1
	Train batch size	256 trajectories/step
	PPO mini-batch size	256
	PPO micro-batch size	1 + dynamic batching
	Entropy coefficient	0.0
	KL regularization	disabled
Distillation		
	Student chunk size	1024
	Teacher chunk size	128
	Teacher max logprobs	16
	Token selection ratio	1.0
	Policy loss mode	reinforce
Advantage and loss		
	Loss max clamp	10

Table 11: Method-specific hyperparameters. Entries marked "none" use the shared configuration unchanged.

Method	Method-specific hyperparameters
OPD	None beyond the shared configuration.
ExOPD	`exopd_lambda`=1.25; `loss_max_clamp`=null; `loss_agg_mode`=token-mean. LoRA: rank=64, alpha=128, target=all-linear.
EOPD	`eopd_entropy_threshold`=0.8; `eopd_alpha`=1.0; teacher max logprobs=17.
Uni-OPD	Batch structure: 16 questions per step, 16 rollouts per question (256 trajectories per step); mini-batch size 16 questions. Loss: `loss_agg_mode`=token-mean; `loss_max_clamp`=null. Margin calibration: `target_correct_ratio`=0.5; `margin_scope`=group; `trajectory_reduce`=mean; `margin_direction`=spread; `margin_delta`=0.4.
Privileged-OPD	Max prompt length=12,288 to accommodate reference solutions; longer solutions are truncated. Otherwise identical to OPD.

both the current policy and the pre-RL base model. To avoid storing duplicate weights, we parameterize the student with LoRA adapters (rank 64, alpha 128, all linear layers) and switch between base and policy by disabling or enabling the adapter. We set `loss_max_clamp`=null and `loss_agg_mode`=token-mean.

EOPD. EOPD augments the reverse-KL distillation objective with a forward-KL term applied selectively to high-entropy teacher tokens (Jin et al., 2026). The entropy threshold $\tau_{\text{ent}} = 0.8$ determines which tokens receive the forward-KL treatment: tokens whose teacher entropy exceeds this threshold are supervised with forward KL to preserve generation diversity, while lower-entropy tokens retain the standard reverse-KL objective. The mixing coefficient $\alpha_{\text{ent}} = 1.0$ controls the weight of the forward-KL term relative to the reverse-KL term. To compute teacher entropy, EOPD requires one additional log-probability slot beyond the top-16 used by other methods; we therefore set teacher max logprobs to 17 for EOPD only.

Uni-OPD. Uni-OPD introduces a dual-perspective recipe that combines student-side data balancing with teacher-side outcome-guided margin calibration (Hou et al., 2026a). On the student side, an online correctness-aware filter reshapes each training batch to maintain a target correct-to-incorrect ratio $\rho_{\text{corr}} = 0.5$; we implement this by training on 16 questions per step with n=16 rollouts per question, yielding 256 trajectories per step, and applying sample filtering to achieve the target ratio. The mini-batch size is correspondingly set to 16 questions. On the teacher side, Uni-OPD calibrates token-level teacher margins against trajectory-level outcome rewards. The margin scope is set to `group`, meaning calibration is performed within each group of rollouts for the same question. The trajectory reduction is `mean`, the margin direction is `spread` (which spreads apart the margins of correct and incorrect trajectories), and the target margin is $\delta_{\text{margin}} = 0.4$. The loss aggregation mode is token-mean, and the loss is not clamped.

Privileged-OPD. Privileged-OPD conditions the teacher on additional training-time information, specifically the verified reference solution for each problem (Ye et al., 2026; Kaur et al., 2026). Because reference solutions are substantially longer than the problem statements, we increase the maximum prompt length from the shared default to 12,288 tokens to accommodate the concatenation of problem, reference solution, and student rollout prefix. Solutions exceeding this length are truncated. All other settings follow the standard OPD configuration. In our implementation, the privileged context is prepended to the problem and student prefix in the teacher's input, while the student receives only the problem and its own rollout, preserving the information asymmetry that Privileged-OPD exploits.

Cal-OPD. Cal-OPD follows the shared configuration above. It additionally uses the evaluative feedback interventions $c_{\text{eval}}^{\text{pos}}$ and $c_{\text{eval}}^{\text{neg}}$ to probe the teacher's self-deviation region, with relaxation factor $\lambda = 5$. These interventions are used only to estimate the TSD region and are not directly distilled into the student. The distillation loss then operates on the calibrated advantage A_t^{Cal} in place of the raw A_t^{OPD}, as described in Section 3.

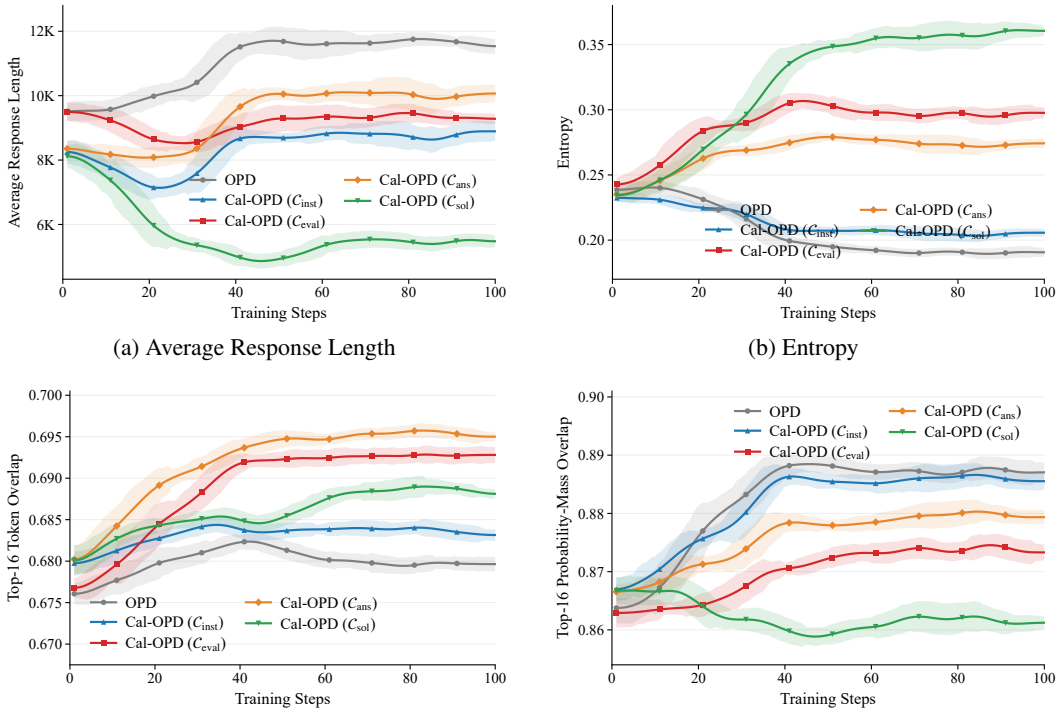
A.7 Additional Training Dynamics

OPD vs. Cal-OPD with evaluative calibration. Figure 9 reveals qualitatively different training dynamics between OPD and Cal-OPD with $\mathcal{C}_{\text{eval}}$. OPD progressively increases response length while reducing student entropy, whereas Cal-OPD maintains substantially shorter responses and higher entropy throughout training. This contrast is also reflected in how each method aligns with the teacher's top-16 distribution. Let H denote the student entropy, $\mathrm{Top16Tok}$ the top-16 token overlap, and $\mathrm{Top16Mass}$ the top-16 probability-mass overlap. The dynamics can be summarized as

$$\begin{aligned} H_{\text{Cal-OPD}} &> H_{\text{OPD}}, \\ \mathrm{Top16Tok}_{\text{Cal-OPD}} &> \mathrm{Top16Tok}_{\text{OPD}}, \\ \mathrm{Top16Mass}_{\text{Cal-OPD}} &< \mathrm{Top16Mass}_{\text{OPD}}. \end{aligned} \tag{17}$$

In particular, the top-16 token overlap under Cal-OPD steadily increases to approximately 0.693, whereas OPD peaks earlier and then slightly declines. Meanwhile, OPD achieves a higher top-16 probability-mass overlap, indicating stronger matching of the teacher's probability allocation.

Together, these dynamics suggest that OPD increasingly fits the teacher's dominant probability pattern, whereas Cal-OPD preserves a broader student distribution while covering more teacher-supported token patterns, enabling the student to capture multiple plausible reasoning modes rather than overfitting to a particular one.



(a) Average Response Length

(b) Entropy

(c) Top-16 Token Overlap

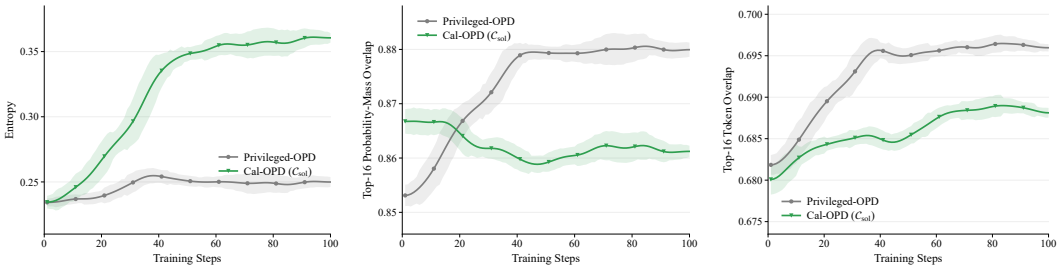
(d) Top-16 Probability-Mass Overlap

Figure 9: Training dynamics of OPD and Cal-OPD on Qwen3-1.7B.

Cal-OPD with instruction-level calibration. In contrast to evaluative calibration, instruction-level calibration with $\mathcal{C}_{\text{inst}}$ yields dynamics that **closely resemble standard OPD**, indicating that its calibration signal is too weak to meaningfully separate TSD from transferable knowledge. As shown in Figure 9, the student entropy under $\mathcal{C}_{\text{inst}}$ follows the same declining trend as OPD and stabilizes around 0.20, whereas the other Cal-OPD variants maintain substantially higher entropy. Its top-16 probability-mass overlap is also nearly identical to that of OPD, stabilizing around 0.886, which shows that it continues to match the teacher's dominant probability allocation as strongly as the uncalibrated objective. Similarly, its top-16 token overlap remains low at approximately 0.684, again close to OPD and well below the levels reached by $\mathcal{C}_{\text{eval}}$, $\mathcal{C}_{\text{ans}}$, and $\mathcal{C}_{\text{sol}}$. These trends indicate that, without task-specific information or strong semantic contrast, the estimated TSD region is too narrow to filter the teacher–student discrepancy effectively, and Cal-OPD largely degenerates to standard OPD behavior.

Cal-OPD with solution-level calibration. Solution-level calibration with $\mathcal{C}_{\text{sol}}$ lies at the opposite extreme: it over-filters the teacher–student discrepancy and causes the supervision signal to collapse. As shown in Figure 9, the student entropy under $\mathcal{C}_{\text{sol}}$ is the highest among all methods and continues to rise throughout training, stabilizing near 0.36, while its average response length is the lowest and steadily decreases to roughly 5.5K tokens. More strikingly, its top-16 probability-mass overlap is not only the lowest among all variants but also falls below its initial value, stabilizing around 0.862. Its top-16 token overlap remains moderate at approximately 0.688, below $\mathcal{C}_{\text{eval}}$ and $\mathcal{C}_{\text{ans}}$, but above OPD. These trends indicate that the TSD region estimated from solution-level privilege is excessively broad, so that a large fraction of the teacher–student discrepancy is classified as TSD and zeroed out. As a result, the student receives too few effective supervision signals to follow the teacher distribution, and instead drifts toward a high-entropy, short-response regime. **Over-calibration therefore degrades the signal more severely than no calibration at all**, which is consistent with its lowest downstream performance among all variants.

Failure Modes of Privileged-OPD and Cal-OPD with Solution-Level Calibration. Both methods shorten responses and degrade performance, but for different reasons. All top-16 metrics are



(a) Entropy (b) Top-16 Probability-Mass Overlap (c) Top-16 Token Overlap

Figure 10: Training dynamics comparison between privileged OPD and Cal-OPD with solution-level calibration.

Table 12: Efficiency comparison between Cal-OPD and OPD across different teacher–student configurations.

Method	**Transfer**	**Avg. time per step (min)**	**Total time for 100 steps (h)**
Cal-OPD	30B-A3B-2507 → 8B	50.05	∼83.4
OPD	30B-A3B-2507 → 8B	35.19	∼58.7
Cal-OPD	30B-A3B-2507 → 4B	48.76	∼81.3
OPD	30B-A3B-2507 → 4B	31.26	∼52.1
Cal-OPD	4B-2507 → 1.7B	14.90	∼24.8
OPD	4B-2507 → 1.7B	18.80	∼31.3

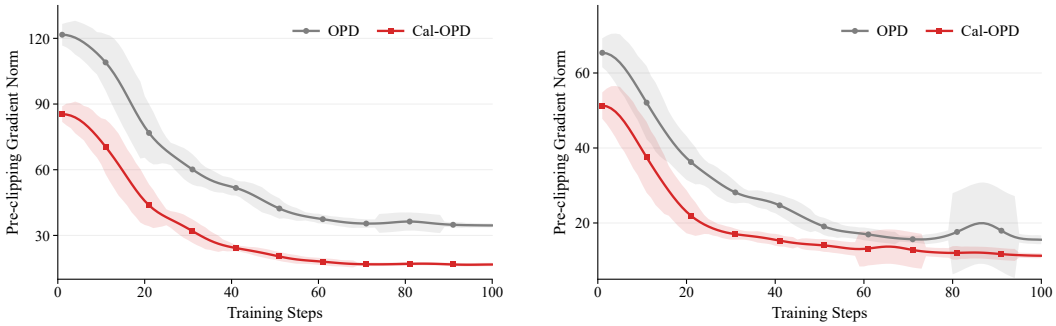
computed against the unprivileged teacher. Privileged-OPD keeps student entropy low and stable (around 0.25) while its top-16 probability-mass overlap rises to about 0.880, indicating that the student becomes overconfident and concentrates on a narrow token set. In contrast, Cal-OPD with $\mathcal{C}_{\mathrm{sol}}$ drives entropy up to about 0.36, and its top-16 probability-mass overlap even drops to roughly 0.859, showing that it fails to match the unprivileged teacher. We attribute this to two distinct failure modes: Privileged-OPD suffers from **overconfident shortcut collapse**, where the student imitates the privileged teacher's sharp distribution but loses diversity; Cal-OPD with $\mathcal{C}_{\mathrm{sol}}$ suffers from **signal collapse**, where an excessively broad TSD region zeros out useful supervision, causing the student to drift into a high-entropy, short-response regime.

A.8 Efficiency Analysis

Efficiency of Cal-OPD. Cal-OPD requires two additional teacher forward passes per step to probe the TSD region, which introduces extra computation. Table 12 compares the wall-clock time per step and the total time for 100 steps between Cal-OPD and OPD across three teacher–student configurations. For the Qwen3-4B-Thinking-2507→Qwen3-1.7B setting, Cal-OPD is actually faster than OPD (14.90 vs. 18.80 minutes per step), because Cal-OPD avoids the response-length expansion that OPD exhibits, and the teacher is not large enough for the extra forward passes to dominate. In the other two configurations, where the teacher (30B-A3B) is substantially larger than the student (8B or 4B), the two additional teacher forward passes become the main computational overhead, making Cal-OPD slower per step (50.05 vs. 35.19 and 48.76 vs. 31.26 minutes). Overall, Cal-OPD remains practically efficient: its extra cost is bounded by two teacher forward passes, and it can even reduce total training time when the teacher–student size gap is moderate.

A.9 Retention-Matched Control on Teacher–Student Discrepancy

Figure 11 reports the pre-clipping gradient norm during OPD training. In both teacher–student configurations, the norm is initially large, around 120 and 65, and gradually decreases to approximately



(a) Qwen3-4B-Thinking-2507 → Qwen3-1.7B (b) Qwen3-30B-A3B-Thinking-2507 → Qwen3-4B

Figure 11: Gradient norm dynamics of OPD and Cal-OPD across different teacher–student pairs.

35 and 15. All methods apply gradient clipping with threshold $C = 1$, so the update is computed as

$$\tilde{\mathbf{g}} = \mathbf{g} \cdot \min\left(1, \frac{C}{\|\mathbf{g}\|}\right). \tag{18}$$

Since $\|\mathbf{g}\|$ is far above 1 throughout training, the raw gradient is scaled by roughly $C/\|\mathbf{g}\|$, corresponding to an attenuation of one to two orders of magnitude. **Thus, substantial signal attenuation is already inherent to standard OPD, and the gains of Cal-OPD cannot be attributed to a smaller overall gradient magnitude. Cal-OPD instead changes the *relative* contribution of individual token advantages by removing the TSD-explained component, while leaving the global clipping behavior unchanged.**

To rule out the possibility that Cal-OPD gains merely from attenuating the optimization signal, we construct three retention-matched baselines that mimic its signal reduction without TSD calibration.

- **Advantage-Sync (Advantage-Level Synchronization).** At each training step k, we read Cal-OPD's advantage retention ratio r_k, defined as the ratio between the sum of absolute calibrated advantages and the sum of absolute original teacher–student discrepancies. We then scale standard OPD's token-level advantages by the same factor, $\tilde{A}_t^{\mathrm{OPD}} = r_k \cdot A_t^{\mathrm{OPD}}$, so that the global advantage magnitude matches Cal-OPD step by step.
- **Token-Sync (Token-Level Synchronization).** At each training step k, we read Cal-OPD's zero-advantage token ratio z_k, i.e., the fraction of tokens whose calibrated advantage is zero. We then randomly select the same fraction z_k of tokens in standard OPD and set their advantages to zero, keeping the remaining advantages unchanged. This matches Cal-OPD's sparsity pattern in terms of the number of discarded tokens, but not which tokens are discarded.
- **TSD-Filter (TSD-Threshold Filtering).** For each token, we compute the maximum absolute deviation induced by the positive and negative privileged interventions, and discard the token entirely if this deviation exceeds a threshold τ_{TSD}. We evaluate three thresholds, $\tau_{\mathrm{TSD}} \in \{0.1, 0.05, 0.01\}$, corresponding to increasingly aggressive filtering. Unlike Cal-OPD, which retains the residual discrepancy beyond the estimated TSD region, this baseline removes the token from the loss altogether.

These controls match Cal-OPD in global advantage scale, token sparsity, and TSD-based token selection, respectively, but omit the calibrated residual.

Table 13 compares Cal-OPD against three retention-matched controls. **Advantage-Sync** applies the same global scaling to OPD's advantages and achieves 50.8, identical to OPD, showing that reducing the overall advantage magnitude alone brings no benefit. **Token-Sync** randomly masks the same fraction of tokens and even drops to 50.1, indicating that matching the sparsity level without selecting the right tokens is insufficient. **TSD-Filter** with $\tau_{\mathrm{TSD}} = 0.1$ reaches 52.7, the strongest control, but still falls short of Cal-OPD; more aggressive thresholds of 0.05 and 0.01 degrade to 51.1 and 49.4, respectively. These results show that neither global attenuation, nor random token removal, nor hard TSD-based filtering can reproduce the gains of Cal-OPD. **The advantage of**

Table 13: Retention-matched control on teacher–student discrepancy for Qwen3-4B-Thinking-2507→Qwen3-1.7B.

Method	**AMC23**	**AIME24**	**AIME25**	**AIME26**	**HMMT26**	**MATH500**	**Avg.**
OPD	82.0	39.6	35.0	34.6	22.9	90.6	50.8
Advantage-Sync	81.7	39.8	35.4	35.0	22.2	90.5	50.8
Token-Sync	81.3	38.3	34.6	34.0	22.3	89.8	50.1
TSD-Filter ($\tau_{\mathrm{TSD}} = 0.1$)	83.6	44.0	37.1	36.0	24.4	90.9	52.7
TSD-Filter ($\tau_{\mathrm{TSD}} = 0.05$)	81.7	41.7	35.4	34.8	22.9	90.1	51.1
TSD-Filter ($\tau_{\mathrm{TSD}} = 0.01$)	80.3	38.8	33.5	32.9	21.6	89.3	49.4
Cal-OPD	84.5	45.0	37.1	36.3	24.8	90.8	**53.1**

Cal-OPD therefore stems from its soft calibration mechanism, which retains the discrepancy beyond the estimated TSD region, rather than from signal attenuation or token sparsity alone.