ERROR-GUIDED SELF-DISTILLATION: TURNING STUDENT FAILURES INTO DISTILLATION SIGNALS

Anonymous authors
Paper under double-blind review

ABSTRACT

Incorrect student attempts contain evidence of how a model fails, yet demonstration-conditioned self-distillation does not explicitly use these failures across attempts as teacher-side context. Simply adding wrong traces is not necessarily helpful: they can mix informative mistakes with irrelevant or misleading reasoning. We introduce Student-Error-Guided Distillation (SEGD), which reuses on-policy student failures as additional privileged information for the teacher. An error-evidence gate selects training examples based on counts of distinct wrong traces and final answers, and an error representation supplies problem-aligned evidence alongside the gold solution. Experiments show improvements on the evaluated mathematical reasoning benchmarks and selected non-math tasks, with mixed broad-transfer results. These findings support investigating student failures as a resource for teacher construction, with selection and representation as central design choices.

1 INTRODUCTION

On-policy distillation (OPD) trains a model on trajectories sampled from its current policy, with a teacher providing token-level supervision on those same trajectories (Agarwal et al., 2024; Gu et al., 2024; Lu & Thinking Machines Lab, 2025). Self-Distillation Fine-Tuning (SDFT) (Shenfeld et al., 2026) instantiates the student and teacher from the same model: the student sees the task input, while the teacher additionally sees an expert demonstration. Training distills this demonstration-conditioned distribution into the student on the student's sampled trajectories.

Incorrect attempts need not be useful as target answers to contain useful information about the student. Correctness-filtered self-training discards failed attempts (Singh et al., 2023), potentially losing evidence of how the student goes wrong. OPD and SDFT already score student-generated trajectories, including incorrect ones, but do not explicitly expose the collected failures across attempts as additional privileged teacher context. Our question is therefore whether these failures can be reused as *diagnostic evidence* for constructing the teacher, rather than as answers for the student to imitate.

This reuse is not automatically beneficial. Wrong attempts may mix informative mistakes with irrelevant reasoning, incomplete solutions, or misleading claims. Simply appending them to a teacher prompt changes the distillation target without ensuring that the change helps learning. In our experiment, raw-error conditioning yielded lower observed answer accuracy than gold-only conditioning (Appendix F). This comparison motivates investigating how to use error evidence more effectively. We therefore study two choices: *which problems should receive error evidence, and how should that evidence be represented?* SEGD uses an error-evidence gate to select examples and an error representation to construct problem-aligned teacher context. Our paired MATH audit describes an association: gate-active examples contain more wrong-trace evidence and a higher share of SEGD-only wins among discordant outcomes (Table 4). This motivates examining error-conditioned teaching. Figure 1 illustrates the conditioning difference, and Appendix E provides concrete examples.

To address this gap, we introduce *Student-Error-Guided Distillation (SEGD)*, which reframes teacher construction in self-distillation: the teacher should not only demonstrate the correct solution, but also respond to the failure modes exposed by the current student. SEGD constructs a failure-aware teacher

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.

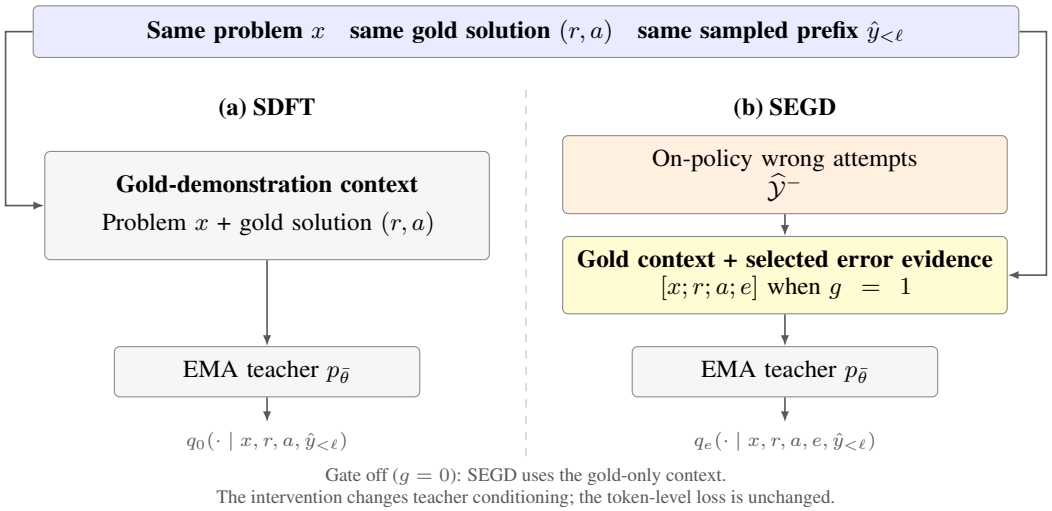

Figure 1: Teacher conditioning in SDFT and SEGD. Both use the same problem, gold solution, and sampled student prefix. With the gate on, SEGD additionally conditions the teacher on a summary of on-policy wrong attempts. With the gate off, it uses the gold-only context. The comparison fixes student and teacher parameters to isolate the context intervention.

by conditioning it on the gold solution together with selected, problem-aligned evidence from the student's on-policy wrong attempts. An *error-evidence gate* selects examples using the number of unique wrong traces and unique wrong answers. These observable counts provide a simple way to prioritize diverse failure evidence without requiring an additional error-type classifier. A compact summary limits the amount of this evidence added to the teacher prompt. The design hypothesis is that problem-specific failures can help the teacher contextualize the gold solution.

Contributions. Our contributions are threefold: (i) we introduce *Student-Error-Guided Distillation (SEGD)*, which uses on-policy student failures as additional privileged teacher context; (ii) we derive how this context changes the reverse-KL gradient; and (iii) we evaluate a practical evidence-count gate and compact error representation through benchmark results, configuration comparisons, and teacher-context diagnostics.

2 Related Work

Knowledge Distillation for LLMs. Knowledge distillation (Hinton et al., 2015) transfers capabilities from teacher to student via soft label matching. Sequence-level distillation (Kim & Rush, 2016) extends this to autoregressive models, and Generalized Knowledge Distillation (GKD) (Agarwal et al., 2024) bridges supervised and on-policy variants. Our work operates entirely within the self-distillation setting, where teacher and student share the same base model.

Privileged-Information Distillation. On-Policy Context Distillation (OPCD) (Ye et al., 2026) provides the teacher with in-context demonstrations invisible to the student. GATES (Stein et al., 2026) uses consensus among document-conditioned tutor responses to gate self-distillation under privileged context. SDFT (Shenfeld et al., 2026) uses expert demonstrations as privileged teacher context; in our reasoning setting, this context is the full gold solution. Privileged Information Distillation (Penaloza et al., 2026) distills training-time privileged information for multi-turn agentic tasks. Our work extends this line by asking what *negative* information, namely the student's own failures, adds when the teacher already has the full solution.

Learning from Errors. Self-training and rejection sampling (Singh et al., 2023) filter incorrect completions out of the training set. DPO (Rafailov et al., 2023) contrasts preferred and dispreferred responses. SEGD instead retains selected incorrect attempts as teacher-side context: its design choices concern which observed failures to expose and how to represent them within an on-policy distillation loop.

3 PRELIMINARIES

Setup. Let $\mathcal{D} = \{(x_i, r_i, a_i)\}_{i=1}^{n}$ be a dataset of problems x_i, gold reasoning traces r_i, and final answers $a_i \in \mathcal{A}$. A completion is a token sequence $y = (y_1, \ldots, y_{|y|}) \in \mathcal{Y}$ over vocabulary $\mathcal{V}$, and the student policy π_θ defines next-token distributions $\pi_\theta(\cdot \mid x, y_{<\ell})$. We use a task verifier $V(y, a) \in \{0, 1\}$ that returns 1 when the completion's final answer matches the gold answer under the task-specific equivalence rule, and 0 otherwise. Starting from a pretrained model and the training triples (x_i, r_i, a_i), our goal is to learn a student policy that answers new problems from x alone; gold traces, verifier outcomes, and sampled student failures are privileged training-time signals used only to construct self-distillation targets.

Token-level on-policy distillation. Following OPD/GKD and SDFT, we write the distillation loss on prefixes visited by the current student rather than on a fixed offline target sequence. Let $p_{\bar{\theta}_t}$ denote the EMA teacher model used for scoring at step t, with $\bar{\theta}_{t+1} = (1-\beta)\bar{\theta}_t + \beta\theta_{t+1}$ after every student optimizer step in the reported EMA training setup. A rollout $\hat{y} \sim \pi_{\theta_t}(\cdot \mid x)$ induces prefixes $(x, \hat{y}_{<\ell})$. For any teacher context u, the token-level KL term is

$$D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid x, \hat{y}_{<\ell}) \,\big\|\, p_{\bar{\theta}_t}(\cdot \mid u, \hat{y}_{<\ell})\big)\,. \tag{1}$$

Here $D_{\mathrm{KL}}(p\|q)$ denotes KL divergence with the argument order shown. Eq. (1) is therefore the reverse-KL direction used in our reported runs, matching the common self-distillation form $D_{\mathrm{KL}}(\pi_\theta(\cdot \mid x) \,\|\, \pi(\cdot \mid x, c))$. The rollout $\hat{y}$, its prefixes, and the teacher distribution are fixed for the update; gradients are taken only through the student distribution π_θ inside the KL. The SEGD teacher construction is independent of this choice: other OPD variants could replace this token KL by a forward-KL or mixed-KL term while keeping the same teacher context u.

Self-Distillation Fine-Tuning (SDFT). In our runs we use EMA rate $\beta = 0.01$. Define the gold-demonstration prompt

$$u^{\mathrm{SDFT}}(x, r, a) = [x;\ r;\ a;\ \text{“Solve correctly.”}], \tag{2}$$

which appends the gold trace and final answer to the problem. The SDFT teacher is therefore the same model conditioned on the demonstration context:

$$q_t^{\mathrm{SDFT}}(\cdot \mid u^{\mathrm{SDFT}}(x, r, a), \hat{y}_{<\ell}) := p_{\bar{\theta}_t}(\cdot \mid u^{\mathrm{SDFT}}(x, r, a), \hat{y}_{<\ell}). \tag{3}$$

The SDFT objective is the on-policy distillation loss with this gold-demonstration context:

$$\mathcal{L}_{\mathrm{SDFT}}(\theta; t) = \mathbb{E}_{(x,r,a)\sim\mathcal{D}}\mathbb{E}_{\hat{y}\sim\pi_{\theta_t}(\cdot|x)}\left[\frac{1}{|\hat{y}|}\sum_{\ell=1}^{|\hat{y}|} D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid x, \hat{y}_{<\ell}) \,\big\|\, q_t^{\mathrm{SDFT}}(\cdot \mid u^{\mathrm{SDFT}}(x, r, a), \hat{y}_{<\ell})\big)\right]. \tag{4}$$

SDFT is demonstration-conditioned: the teacher observes the sampled student prefix, but its privileged context (r, a) does not explicitly summarize failure patterns across the student's attempts.

4 STUDENT-ERROR-GUIDED DISTILLATION

SEGD augments the gold-demonstration teacher context with selected on-policy student failures. Figure 2 shows the error set, gate, and error representation; the token-level distillation objective is unchanged.

What is selected, and what changes? The gate selects *training examples*, not individual tokens or a best wrong answer. Each example's observed wrong attempts determine its evidence score; examples passing the within-generation-batch threshold receive additional error context. The representation then maps their wrong attempts into teacher-side evidence. The student still sees only the problem, and the gold solution remains in the teacher prompt. SDFT already learns on student-generated prefixes, including incorrect ones; SEGD additionally exposes evidence from the sampled attempts on the same problem to the teacher scoring those prefixes. The evidence pool can include the trajectory being scored; it is constructed from complete attempts before prefix scoring. Wrong completions are therefore *context for constructing targets*, not supervised targets to imitate. The verifier identifies incorrect final answers to construct the wrong-attempt set.

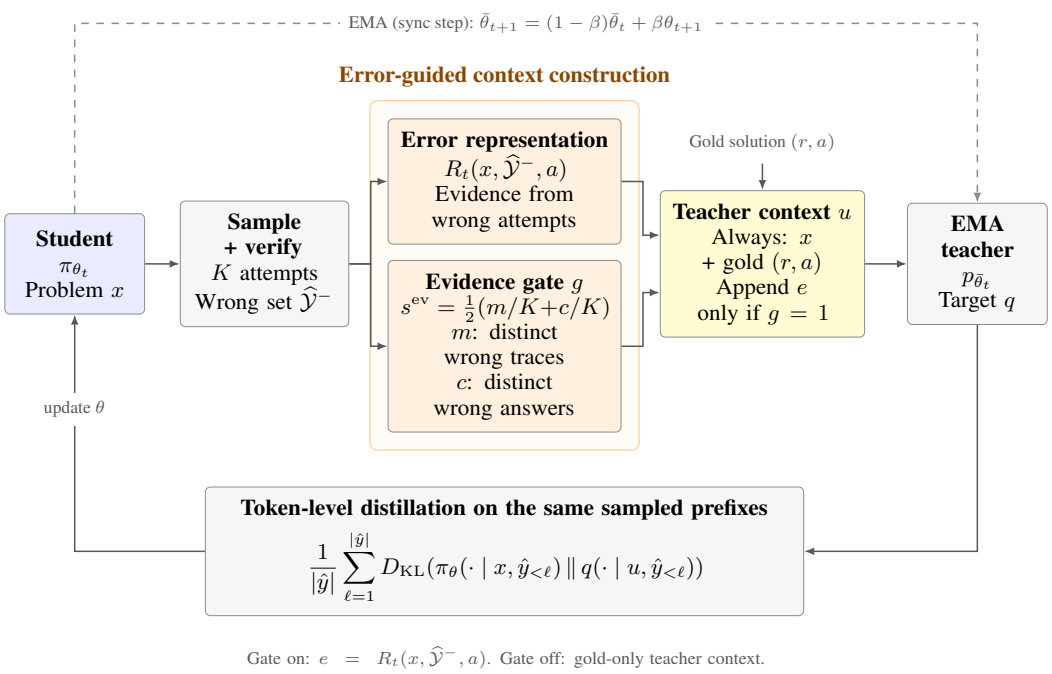

Figure 2: SEGD training pipeline. Verified wrong attempts supply both the error representation and the gate's evidence counts. The gate controls whether this evidence is appended to the gold context. The EMA teacher scores the sampled student prefixes for reverse-KL distillation; the resulting update changes the student parameters. Representation construction is specified in Appendix C.

4.1 A LEARNING-DYNAMICS MOTIVATION

We specialize the per-step learning-dynamics perspective of Ren & Sutherland (2025) to teacher-context interventions under reverse-KL distillation. Fix the student parameters, sampled prefixes, teacher parameters, and averaging weights. Let $p = \text{softmax}(z)$ be the student distribution at one prefix, and let q_0 and q_e be the detached teacher distributions under gold-only and error-conditioned contexts. For $D_{\text{KL}}(p\|q)$, the exact logit gradient is

$$G_j(p, q) := \frac{\partial D_{\text{KL}}(p\|q)}{\partial z_j} = p_j \left[\log \frac{p_j}{q_j} - D_{\text{KL}}(p\|q)\right]. \tag{5}$$

Writing $h_j = \log(q_{e,j}/q_{0,j})$, we have $D_{\text{KL}}(p\|q_e) - D_{\text{KL}}(p\|q_0) = -\mathbb{E}_{v\sim p}[h_v]$. Subtracting the two gradients therefore gives

$$\delta G_j = G_j(p, q_e) - G_j(p, q_0) = -p_j \left[h_j - \mathbb{E}_{v\sim p} h_v\right]. \tag{6}$$

Thus error conditioning changes the gradient through the teacher's relative preference changes, centered and weighted by the student distribution. This identity connects student-specific failure evidence in the teacher context to the student gradient.

To connect this change to learning, let $F(\theta)$ be a smooth evaluation loss, J_s the student-logit Jacobian at prefix s, and α_s fixed averaging weights. Let g_e and g_0 be the corresponding student-parameter loss gradients. Their difference is $\delta g = g_e - g_0 = \sum_s \alpha_s J_s^\top \delta G_s$. For matched ordinary SGD updates from the same parameters,

$$F(\theta - \eta g_e) - F(\theta - \eta g_0) = -\eta \nabla F(\theta)^\top \delta g + O(\eta^2), \tag{7}$$

under bounded local curvature. Helpful influence requires alignment with the evaluation-loss gradient. For AdamW, the optimizer state and transformation also matter; actual update displacements must be used rather than treating the raw gradient as the update. This local analysis of a smooth evaluation loss motivates two empirical design questions: which failures should change the teacher context, and how should those failures be represented? Our gate uses distinct-error counts as a simple selection

heuristic; the representation determines what error evidence is exposed to the teacher. Both act upstream of the teacher preference change h; neither computes the evaluation-gradient alignment above. The derivation therefore motivates error-conditioned teaching, while the experiments assess these particular selection and representation choices.

4.2 On-Policy Error Set, Gate, and Error Representation

At step t, we sample on-policy attempts from the current rollout policy π_{θ_t} and score teacher targets with the EMA teacher $p_{\bar{\theta}_t}$; all error-evidence quantities below are computed from these sampled attempts; we suppress the time subscript on their sets and counts. For a generation batch with problem index set $\mathcal{B}_t$, write each example as (x_i, r_i, a_i). For each $i \in \mathcal{B}_t$, we sample K independent student attempts,

$$\hat{y}_{i,k} \sim \pi_{\theta_t}(\cdot \mid x_i), \qquad k = 1, \ldots, K. \tag{8}$$

We identify the student's observed failures with the verifier:

$$\begin{aligned} \mathcal{K}_i^- &= \{k : V(\hat{y}_{i,k}, a_i) = 0\}, \\ \widehat{\mathcal{Y}}_i^- &= \{\nu(\hat{y}_{i,k}) : k \in \mathcal{K}_i^-\}. \end{aligned} \tag{9}$$

Here ν normalizes completions and removes end-of-sequence tokens; the set retains each normalized completion once. Let $m_i = |\widehat{\mathcal{Y}}_i^-|$ denote the number of unique wrong traces. Let $\widetilde{\mathcal{A}}_i^-$ be the multiset of verifier-normalized wrong final answers produced by the sampled wrong completions, and let $\mathcal{A}_i^- = \mathrm{Unique}(\widetilde{\mathcal{A}}_i^-)$ with $c_i = |\mathcal{A}_i^-|$.

Error-evidence gate. The gate prioritizes examples with more distinct observed failures. Given the unique wrong-trace count m_i and the distinct wrong-answer count c_i, we define

$$s_i^{\mathrm{ev}} = \frac{1}{2}\left(\frac{m_i}{K} + \frac{c_i}{K}\right), \tag{10}$$

Since both counts are computed from K sampled completions, $s_i^{\mathrm{ev}} \in [0, 1]$; it is larger when the current student produces multiple distinct wrong traces and wrong final answers on the same problem. Let $\mathcal{I}_t^- = \{i \in \mathcal{B}_t : m_i > 0\}$ be the examples with at least one observed wrong trace. If $\mathcal{I}_t^- = \varnothing$, we set $g_{i,t} = 0$ for all i. Otherwise, we compute the within-generation-batch threshold

$$\tau_t(\gamma) = \mathrm{Quantile}_\gamma\left((s_j^{\mathrm{ev}})_{j \in \mathcal{I}_t^-}\right) \tag{11}$$

and activate the error channel for examples above this threshold:

$$g_{i,t} = \mathbf{1}\left[i \in \mathcal{I}_t^-\right]\, \mathbf{1}[s_i^{\mathrm{ev}} \geq \tau_t(\gamma)]\,. \tag{12}$$

Here $\mathrm{Quantile}_\gamma$ uses linear interpolation, as in `numpy.percentile`, on the score list with repeated values retained. The threshold is computed over error-containing examples in the current generation batch $\mathcal{B}_t$, targeting its top $(1-\gamma)$ fraction. Since the implementation uses the inclusive condition $s_i^{\mathrm{ev}} \geq \tau_t(\gamma)$, ties at the boundary are included and the realized activation rate can be substantially larger when scores tie; if all scores tie, every error-containing example passes. The threshold adapts to the observed evidence within each batch. Section 6 examines its selection behavior.

Comparator confidence, difficulty, and disagreement scores use the same percentile rule; their definitions are in Appendix A.

Error representation. Let $R_t(x_i, \widehat{\mathcal{Y}}_i^-, a_i)$ construct teacher-side evidence from the observed wrong attempts, with the problem and gold answer available as context. The gate controls whether this representation enters the teacher prompt:

$$e_{i,t} = \begin{cases} R_t(x_i, \widehat{\mathcal{Y}}_i^-, a_i), & g_{i,t} = 1, \\ \varnothing, & g_{i,t} = 0. \end{cases} \tag{13}$$

The representation is held fixed during the distillation update; gradients do not flow through its construction. Our default implementation uses a compact generated summary, while the Raw+Evidence control uses a representative wrong trace; Appendix C specifies both constructions and the prompts. Thus gate-off examples retain the SDFT teacher context, while gate-on examples add problem-aligned evidence of the student's observed failures.

SEGD teacher and loss. SEGD forms the teacher prompt by augmenting the SDFT gold-demonstration prompt only when the error-evidence gate is active:

$$u_{i,t}^{\mathrm{SEGD}} = \begin{cases} \mathrm{Tmpl}(x_i, r_i, a_i, e_{i,t}) & \text{if } g_{i,t} = 1, \\ u^{\mathrm{SDFT}}(x_i, r_i, a_i) & \text{if } g_{i,t} = 0. \end{cases} \tag{14}$$

Here Tmpl denotes the error-conditioned prompt template reproduced in Appendix C, including its final instruction. The corresponding EMA teacher distribution on a sampled prefix is

$$q_{i,t}^{\mathrm{SEGD}}(\cdot \mid u_{i,t}^{\mathrm{SEGD}}, \hat{y}_{i,k,<\ell}) := p_{\bar{\theta}_t}(\cdot \mid u_{i,t}^{\mathrm{SEGD}}, \hat{y}_{i,k,<\ell}). \tag{15}$$

For a sampled rollout $\hat{y}_{i,k}$, define the SEGD token-KL loss

$$\ell_{\mathrm{SEGD}}(\theta; i, k, t) = \frac{1}{|\hat{y}_{i,k}|} \sum_{\ell=1}^{|\hat{y}_{i,k}|} D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid x_i, \hat{y}_{i,k,<\ell}) \,\big\|\, q_{i,t}^{\mathrm{SEGD}}(\cdot \mid u_{i,t}^{\mathrm{SEGD}}, \hat{y}_{i,k,<\ell})\big)\,. \tag{16}$$

Because the threshold in Eq. (12) is computed within $\mathcal{B}_t$, the unweighted stochastic objective is a mini-batch average:

$$\widehat{\mathcal{L}}_{\mathrm{SEGD}}(\theta; t, \mathcal{B}_t, \widehat{Y}_t) = \frac{1}{|\mathcal{B}_t|} \sum_{i \in \mathcal{B}_t} \frac{1}{K} \sum_{k=1}^{K} \ell_{\mathrm{SEGD}}(\theta; i, k, t), \tag{17}$$

where the rollouts are sampled from $\pi_{\theta_t}^{\otimes K}(y_{1:K} \mid x_i) := \prod_{k=1}^{K} \pi_{\theta_t}(y_k \mid x_i)$ and $g_{i,t}$, $e_{i,t}$, and $u_{i,t}^{\mathrm{SEGD}}$ are computed after scoring the whole batch. The corresponding population objective averages over both batches and rollouts:

$$\mathcal{L}_{\mathrm{SEGD}}(\theta; t) := \mathbb{E}_{\mathcal{B}_t} \mathbb{E}_{\widehat{Y}_t \sim P_t(\cdot \mid \mathcal{B}_t)}[\widehat{\mathcal{L}}_{\mathrm{SEGD}}(\theta; t, \mathcal{B}_t, \widehat{Y}_t)]. \tag{18}$$

Here $\mathcal{B}_t$ indexes the problems in a generation batch drawn from $\mathcal{D}$, $\widehat{Y}_t = (\hat{y}_{i,k})_{i \in \mathcal{B}_t,\, k=1}^{K}$, and $P_t(\widehat{Y}_t \mid \mathcal{B}_t) = \prod_{i \in \mathcal{B}_t} \prod_{k=1}^{K} \pi_{\theta_t}(\hat{y}_{i,k} \mid x_i)$. The rollout argument in Eq. (17) determines the gate and teacher context. All sampled quantities are held fixed when differentiating with respect to θ. Appendix C specifies the masks and weights used in optimization. When $g_{i,t} = 0$, Eq. (14) reduces to the SDFT teacher prompt, so the teacher context and token-level KL term match SDFT on that example at the same fixed parameters and prefixes. This pointwise equality does not imply equality of separately trained models or of rollout expectations conditioned on the data-dependent gate. When $g_{i,t} = 1$, the gold solution remains present and uncompressed, while the selected student-error context provides additional teacher-side information about the current student's observed failures. Thus SEGD changes the teacher construction while retaining the token-level KL term of Eq. (4) and the EMA update described in the setup.

5 Experiments

Model and data. We use Qwen2.5-7B-Instruct (Yang et al., 2024) as the student/teacher and fine-tune LoRA adapters (Hu et al., 2021) ($r = 128$). Training uses the canonical Hendrycks MATH split (7,500 train / 5,000 test) (Hendrycks et al., 2021). Primary evaluation uses the full MATH test set and MATH-500; secondary evaluation uses multi-seed GSM8K (Cobbe et al., 2021). We also evaluate broad-transfer benchmarks, including MMLU-Pro (Wang et al., 2024), SocialIQA (Sap et al., 2019), and CommonsenseQA (Talmor et al., 2019) in Table 1. For direct non-math SEGD evidence, we run a 3-seed same-domain MC-QA study on CommonsenseQA, ARC-Challenge (Clark et al., 2018), and OpenBookQA (Mihaylov et al., 2018) with Qwen2.5-3B (Table 3). Separately, we include a 3-seed ToolUse pilot with clean and corrupted demonstrations as auxiliary evidence about the broader setup. The on-policy math runs generate $K = 12$ student attempts per problem via vLLM (Kwon et al., 2023).

Baselines. Table 1 compares against four trainable 7B baselines covering standard supervised tuning, self-distillation, and error-centric alternatives: supervised fine-tuning (SFT) on gold solutions; SDFT (Shenfeld et al., 2026); ERRDPO, which treats the teacher trace as chosen and the student wrong trace as rejected; and STEPKL, which applies the distillation loss only after the first localized divergence from the teacher trace. Zero-shot instruction-model references from the same model family

Table 1: Main 7B math and broad-transfer results. Scores are percentages; trainable rows report mean ± standard deviation over three seeds. Bold scores indicate the highest mean among trainable methods in each column. A † denotes Holm-adjusted $p < 0.05$ in a descriptive pooled paired Wilcoxon check (Appendix Table 5).

	In-domain		Transfer			
Method	**MATH**	**MATH-500**	**GSM8K**	**MMLU-Pro**	**SocialIQA**	**CSQA**
Zero-shot instruction-model references (single run; not trainable baselines)						
Qwen2.5-7B-Instruct	64.14	61.40	79.15	56.39	63.62	74.94
Qwen2.5-14B-Instruct	63.04	62.80	81.35	63.31	72.66	79.28
Qwen2.5-32B-Instruct	72.80	72.40	83.24	69.38	70.23	75.10
DeepSeek-R1-Distill-Qwen-7B	62.96	60.60	76.57	68.91	69.28	67.44
Trainable fine-tuning and distillation methods						
SFT	48.34 ± 0.16	47.20 ± 1.25	79.66 ± 0.27	55.59 ± 0.12	62.63 ± 0.18	62.06 ± 0.25
SDFT	62.67 ± 0.54	63.80 ± 1.60	83.83 ± 0.31	53.32 ± 0.16	63.39 ± 3.49	63.96 ± 5.26
ERRDPO	63.25 ± 0.18	65.40 ± 0.72	82.46 ± 0.96	54.59 ± 0.19	**65.81 ± 1.11**	68.52 ± 2.89
STEPKL	62.64 ± 0.37	65.27 ± 0.61	83.09 ± 1.07	54.31 ± 0.18	64.41 ± 2.57	67.46 ± 3.45
SEGD	**69.00 ± 0.19**†	**70.33 ± 0.64**†	**84.00 ± 0.20**	**56.03 ± 0.26**	65.41 ± 1.14	**72.48 ± 0.43**

and one external 7B reasoning model are included as scale/context anchors, not as trainable-method baselines. Table 3 reports auxiliary same-domain checks in settings where our training/evaluation pipeline supports SFT, SDFT, and either SEGD or an EMA self-distillation pilot.

Evaluation. Our primary metric is the robust verifier accuracy, which checks symbolic equivalence of boxed answers rather than exact string match. Trainable-method comparisons use 3 random seeds; zero-shot instruction-model rows are single-run references. The MC-QA study uses exact answer-letter accuracy, while ToolUse uses exact tool-call accuracy. We report mean ± standard deviation where multiple seeds are available.

Table 1 evaluates SEGD as error-guided teacher construction: the teacher receives the gold solution together with selected evidence of the student's own failures. On the math benchmarks, SEGD consistently outperforms SDFT and the direct error-centric baselines ERRDPO and STEPKL. Descriptive pooled paired Wilcoxon tests on saved correctness vectors further support the MATH and MATH-500 gains after Holm correction (Appendix Table 5). The broad-transfer results are intentionally mixed: SEGD is strongest among trainable rows on MMLU-Pro and CommonsenseQA, while ERRDPO is slightly strongest on SocialIQA.

The same-family zero-shot rows contextualize scale: SEGD improves over the corresponding instruction-model reference on the main reasoning metrics, while larger instruction models remain stronger on several transfer columns.

Gate and error-representation ablations. To examine sensitivity to the selection rule, we compare the error-evidence score with three alternative gates using the same percentile rule: a confidence gate based on the student's mean log-probability, a difficulty gate based on the observed error rate, and a disagreement gate based on how much more likely wrong attempts are under the student than under the gold-demonstration teacher. These comparisons assess alternative selection heuristics. Their interpretation also depends on intervention exposure, training endpoint, and seed coverage.

Table 2 reports sensitivity across gate and representation configurations. The raw-context configuration scores above the alternative-gate and shuffled-summary configurations.

Table 2: Gate and error-representation configurations. Scores are percentages. The final row repeats SEGD from Table 1. The disagreement gate uses one seed; all other rows use three.

Gate / Error representation	MATH	MATH-500
Summary + confidence gate	67.90	68.33
Summary + disagreement gate	67.52	67.60
Summary + difficulty gate	67.97	67.73
Summary + entropy-proxy gate ($\hat{\rho}_i \geq 1/2$, $\hat{H}_i \leq \log 2$)	65.61	67.07
Shuffled summary + error-evidence gate	65.85	67.80
Raw + error-evidence gate (no summary)	68.19	68.93
Summary + error-evidence gate (SEGD)	**69.00**	**70.33**

Summary representation. Raw+Evidence is configured to replace the summary with a representative wrong trace. We use a summary to limit added context. A separate audit of six archived raw/summary runs found zero logged intervention exposure.

Worked success case: recovering a missing degree of freedom. The sequence problem in Appendix E asks for a_1 when the second differences are all 1 and $a_{19} = a_{92} = 0$. One recorded error pattern assumes $a_n = a_1 + (n-1)n/2$, which silently fixes the initial first difference and yields $a_1 = -171$ from the first constraint. The missing freedom is visible in the correct form: constant second difference 1 gives a quadratic with leading coefficient $1/2$, so the two roots imply $a_n = \frac{1}{2}(n-19)(n-92)$ and hence $a_1 = 819$. The reported SDFT output is -171, whereas SEGD outputs 819.

The appendix's abridged error representation identifies the restrictive sequence form and errors in solving the two constraints. Under the SEGD construction, such evidence supplements the full gold solution only for examples passing the gate; it tells the teacher what went wrong across attempts while preserving the correct derivation as context. This illustrates the intended use of wrong attempts as diagnostic information. The case presents reported outputs and an abridged error representation. Separate gate-on and gate-off examples in Appendix E illustrate the selection behavior.

6 Analysis

What the error-evidence gate measures. The score combines distinct wrong-trace and wrong-answer counts using a within-generation-batch threshold. It reflects both the frequency and diversity of observed failures: when wrong traces are all distinct, its first term equals the fraction of incorrect attempts. String-level distinctness need not imply distinct reasoning patterns.

In the paired 100-example MATH audit (Appendix Table 4), gate-active examples contain more distinct wrong traces. Among examples where exactly one method is correct, the SEGD-only share is higher in the gate-active group. This descriptive association links gate activity with the observed error evidence and outcomes.

The entropy-proxy comparator instead requires a dominant wrong answer ($\hat{\rho}_i = \max_a \hat{p}_i(a) \geq 1/2$) and low wrong-answer entropy ($\hat{H}_i = -\sum_a \hat{p}_i(a) \log \hat{p}_i(a) \leq \log 2$), where $\hat{p}_i$ is the empirical distribution over wrong answers. Only 3–9% of wrong chunks pass in its active phase. This highlights the different selection coverage of the two gate designs.

The same-domain MC-QA study in Table 3 shows that error-guided teacher conditioning can help when the verifier is an answer-letter checker rather than symbolic math equivalence. It also exposes useful nuance: SEGD is very strong on ARC-Challenge and modestly best on OpenBookQA, while CommonsenseQA shows limited gains and is slightly better under SFT.

Training cost is small. At 7B, the full error-guided phase adds approximately 7.4 minutes over a two-hour SDFT run ($1.06\times$). At 3B, the overhead is $1.02\times$. In the compact-summary implementation, summary generation accounts for almost all of the overhead; the gate computation is negligible. The implementation generates compact summaries during the active phase for chunks with observed

Table 3: Auxiliary 3B same-domain checks. Top: direct non-math SEGD evidence on multiple-choice QA, where training and evaluation use the same task family. Bottom: a ToolUse pilot for EMA self-distillation under clean and corrupted demonstrations; this supports the broader self-distillation setup but is not used as a direct SEGD headline result. Scores are percentages over three seeds.

3B same-domain MC-QA			
Method	**CSQA**	**ARC-C**	**OBQA**
SFT	**79.03 ± 0.14**	13.38 ± 1.46	74.87 ± 0.42
SDFT	78.00 ± 0.97	36.12 ± 3.62	76.53 ± 0.50
SEGD	78.35 ± 0.72	**79.82 ± 1.96**†	**79.13 ± 0.23**

3B same-domain ToolUse			
Method	**Clean**	**20% noisy**	**40% noisy**
SFT	75.49 ± 1.70	73.04 ± 0.85	71.57 ± 2.25
SDFT	76.96 ± 0.85	76.47 ± 1.47	74.51 ± 3.06
EMA self-distill	**81.37 ± 1.70**	**79.41 ± 2.55**	**75.00 ± 5.30**

wrong traces and then injects them only when the gate passes, so the cost is driven by active-phase wrong-trace incidence rather than by the final gate-on fraction alone.

7 Conclusion

SEGD introduces on-policy student failures as additional teacher-side context for self-distillation, using an error-evidence gate and a compact error representation while retaining the token-level distillation objective. The reported reasoning results motivate this approach, while transfer and configuration comparisons show that its performance depends on the setting. Our learning-dynamics analysis connects teacher-context changes to student gradients. Future work can develop more precise evidence selection and faithful error representations across a broader range of tasks.

AI Use Statement

Generative AI tools, including ChatGPT/Codex, assisted with hypothesis and experiment design, mathematical derivations and their explanation, implementation and debugging, interpretation of experimental results, and manuscript drafting and revision. The authors are responsible for the final text, claims, references, code, and results.

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 *The twelfth international conference on learning representations*, 2024.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *The twelfth international conference on learning representations*, 2024.

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

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

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.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

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

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2381–2391, 2018.

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *International Conference on Learning Representations*, 2025.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *EMNLP*, 2019. URL https://www.aclweb.org/anthology/D19-1454.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, et al. Beyond human data: Scaling self-training for problem-solving with language models. *arXiv preprint arXiv:2312.06585*, 2023.

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

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4149–4158, 2019.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

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

A Comparator Gate Definitions

Table 2 keeps the same percentile rule, error-containing reference set $\mathcal{I}_t^-$, and inclusive thresholding convention from Eq. (12), but swaps in alternative scores. The confidence gate uses the student's mean token log-probability across sampled attempts,

$$s_i^{\text{conf}} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\hat{y}_{i,k}|}\sum_{\ell=1}^{|\hat{y}_{i,k}|}\log \pi_{\theta_t}(\hat{y}_{i,k,\ell} \mid x_i, \hat{y}_{i,k,<\ell}). \tag{19}$$

The same top-quantile rule therefore selects relatively high-confidence examples; it is a confidence comparator rather than an uncertainty gate. The difficulty gate uses the observed error rate,

$$s_i^{\text{diff}} = \frac{|\mathcal{K}_i^-|}{K}. \tag{20}$$

The disagreement gate compares each wrong attempt against the gold-demonstration SDFT teacher,

$$s_i^{\text{dis}} = \frac{1}{|\mathcal{K}_i^-|}\sum_{k\in\mathcal{K}_i^-}\max\left\{0, \bar{\ell}_{\theta_t}(\hat{y}_{i,k} \mid x_i) - \bar{\ell}_{\bar{\theta}_t}^{\text{SDFT}}(\hat{y}_{i,k} \mid x_i, r_i, a_i)\right\}, \tag{21}$$

where $\bar{\ell}_{\theta_t}(\hat{y}_{i,k} \mid x_i)$ is the mean token log-probability of $\hat{y}_{i,k}$ under the student and $\bar{\ell}_{\bar{\theta}_t}^{\text{SDFT}}(\hat{y}_{i,k} \mid x_i, r_i, a_i)$ is the corresponding mean token log-probability under the gold-demonstration EMA teacher $p_{\bar{\theta}_t}(\cdot \mid u^{\text{SDFT}}(x_i, r_i, a_i), \hat{y}_{i,k,<\ell})$. These controls test whether generic confidence, raw difficulty, or student–teacher mismatch can replace observed student-error evidence.

B Broader Impact

Error-guided self-distillation could make post-training more data-efficient by reusing a model's own failures as targeted training evidence, reducing dependence on additional human annotation for domains with reliable verifiers. This is most beneficial in educational, scientific, and formal reasoning settings where correcting recurring mistakes is valuable and outputs can be checked. The same idea also carries risks: if the verifier is weak, biased, or misaligned with the intended task, SEGD may reinforce spurious corrections; if error summaries are used without gating or auditing, they may amplify model-generated misconceptions rather than repair them. We therefore view error-guided distillation as appropriate for settings with transparent evaluation, careful monitoring of failure cases, and human oversight before deployment in high-stakes applications.

C Implementation Details

Default error representation. For a nonempty wrong set, the default representation in Eq. (13) is a summary generated by the current rollout model:

$$R_t^{\text{sum}}(x_i, \widehat{\mathcal{Y}}_i^-, a_i) = \text{GreedyGen}_{\pi_{\theta_t}}\left(v(x_i, \widehat{\mathcal{Y}}_i^-, a_i); L_c\right). \tag{22}$$

The prompt v lists the incorrect attempts and gold answer and requests a concise error summary. This is a choice of representation, not an additional training objective or a requirement of the teacher-conditioning formulation.

Summary prompt. For each problem, we collect the unique incorrect attempts after stripping end-of-sequence tokens. During the active phase, if the incorrect set is non-empty, we form the summary prompt:

```
Problem: <student prompt>
-- Incorrect Attempt 1 --
<trace>
...
Correct Answer: <a>
Write a concise error summary that captures the main
mathematical or logical mistakes across the incorrect
attempts above.
Start with 'Error summary:'
```

and decode a single summary with temperature 0 and summary max length $L_c = 128$ using the current rollout model. The summary is generated before teacher scoring; the EMA teacher is used to score the final augmented prompt, not to write the summary. Summaries for gate-off problems are generated but not injected.

Teacher prompt injection. When the error-evidence gate activates, the compact-summary teacher prompt appends:

```
Error summary across the incorrect attempts:
<summary>
Now solve this problem correctly with your own reasoning,
avoiding the mistake shown above.  Put your final answer
within \boxed{}.
```

Otherwise it falls back to the SDFT gold-demonstration prompt with the full gold solution. The Raw+Evidence control uses the same gate but appends the first unique wrong trace for that problem as `Wrong attempt: <trace>` instead of the summary. The optional aggregate-teacher path can average distributions over multiple raw-trace prompts, but it is not used for the reported main 7B rows.

Hyperparameters. LoRA rank $r_{\text{LoRA}} = 128$, learning rate 5×10^{-6}, batch size 16, EMA rate $\beta = 0.01$, $K = 12$ on-policy samples, gate percentile $\gamma = 0.7$, and reverse-KL distillation. The implementation masks truncated completions in the main math runs, applies a vLLM token-ratio correction to student rollouts, and skips the first three completion tokens in the distillation loss. Three-seed training runs use random seeds $\{42, 123, 3407\}$; single-seed comparisons and single-run evaluations are identified separately in the results.

Distillation divergence. The KL term in Eq. (1) is computed over the full vocabulary in the reported 7B runs:

$$D_{\text{KL}}(\pi_\theta \| p_{\bar{\theta}_t}) = D_{\text{KL}}\big(\pi_\theta(\cdot \mid x, \hat{y}_{<\ell}) \,\big\|\, p_{\bar{\theta}_t}(\cdot \mid u, \hat{y}_{<\ell})\big)\,. \tag{23}$$

This follows the OPD/SDFT reverse-KL direction, with sampled prefixes and teacher probabilities held fixed during the update. The sampled-token log-ratio view used in OPD/SDFT remains useful intuition, but our implementation computes the full-vocabulary token KL rather than a sampled-token estimator.

Batch terminology. A generation batch contains problem groups, each with K attempts; the code calls each such group a chunk. The gate reference set contains the error-producing problems in that generation batch. The optimizer batch size reported above is a separate training setting and does not define the number of problems in the gate reference set.

Masked and weighted implementation. For rollout (i, k), let $M_{i,k,\ell}$ indicate a non-padding token with $\ell > 3$ in a non-truncated completion, and set $Z_{i,k} = \max(1, \sum_\ell M_{i,k,\ell})$. Let $d_{i,k,\ell}(\theta)$ denote the full-vocabulary token KL in Eq. (16). The vLLM correction uses detached token ratios

$$\rho_{i,k,\ell} = \min\Big\{C, \exp\big(\log p^{\text{old}}_{i,k,\ell} - \log p^{\text{sample}}_{i,k,\ell}\big)\Big\}\,, \qquad w_{i,k} = \frac{\sum_\ell M_{i,k,\ell}\rho_{i,k,\ell}}{Z_{i,k}}, \tag{24}$$

where the probabilities score the sampled token at its prefix under the training-model snapshot and the sampling engine, respectively, and C is the configured ratio cap. Thus the correction is averaged over retained tokens and applied as a heuristic sequence weight, rather than the product of token ratios that defines a sequence-level importance ratio. With correction disabled, $w_{i,k} = 1$. The shared SDFT/SEGD loss routine applies these masks and weights independently of the teacher-context choice. With matched masking and correction settings, SDFT uses the same expression with its gold-only teacher distribution. For a local loss batch with problem index set $\mathcal{J}$, the standard SEGD path uses

$$\widehat{\mathcal{L}}^{\text{impl}} = \frac{1}{|\mathcal{J}|K} \sum_{i \in \mathcal{J}} \sum_{k=1}^{K} \frac{w_{i,k}}{Z_{i,k}} \sum_\ell M_{i,k,\ell} d_{i,k,\ell}(\theta). \tag{25}$$

Fully masked sequences contribute zero and remain in the sequence-average denominator. The trainer divides this loss by the current gradient-accumulation factor before backpropagation. Teacher synchronization runs in the step-end callback after every optimizer step (`ref_model_sync_steps=1`), using the updated student parameters.

Table 4: Error-evidence bucket audit on 100 paired seed-42 MATH examples. SEGD-only means SEGD is correct and SDFT is wrong; SDFT-only means the reverse. The gate-active bucket has more wrong-trace evidence and a much higher SEGD share among one-method-correct flips. Evidence scores are scaled by 100 for display.

Evidence bucket	n	SEGD-only	SDFT-only	Net	SEGD flip %	$100s^{\text{ev}}$	Unique wrong
Gate off / low evidence	72	27	18	+9	60.00	25.50	5.35
Gate on / high evidence	28	13	2	+11	86.67	61.60	11.14

Table 5: Selected Wilcoxon checks for highlighted cells in Tables 1 and 3. Each row compares the highlighted method against the strongest listed trainable reference for that column, or the nearest reference when the highlighted method is not best. The 7B and MC-QA rows use pooled paired seed-example correctness vectors and should be read as descriptive because repeated seeds share benchmark items. ToolUse results are reported separately without inferential tests. Δ is candidate accuracy minus reference accuracy in percentage points.

Group	Metric	Reference	Pairs	Δ	p_{one}	Holm p	Readout
7B main	MATH	ERRDPO	15000	+5.75	1.58×10^{-34}	3.33×10^{-33}	$p_{\text{Holm}} < 0.05$
7B main	MATH-500	ERRDPO	1500	+4.93	4.64×10^{-4}	6.50×10^{-3}	$p_{\text{Holm}} < 0.05$
7B main	GSM8K	SDFT	3957	+0.18	4.20×10^{-1}	1.00	$p_{\text{Holm}} \geq 0.05$
MC-QA	CSQA	SFT	3663	-0.68	7.66×10^{-1}	7.66×10^{-1}	SFT higher
MC-QA	ARC-C	SDFT	897	+43.70	5.80×10^{-70}	2.90×10^{-69}	$p_{\text{Holm}} < 0.05$
MC-QA	OBQA	SDFT	1500	+2.60	3.73×10^{-2}	1.12×10^{-1}	uncorrected only

D Descriptive Paired Statistical Checks

For Table 1, we compute paired Wilcoxon signed-rank tests on saved per-example correctness vectors. The pairing unit is a seed-example pair: the same benchmark item is compared under SEGD and a reference method within each matched seed, then pooled across seeds. This gives a sensitive paired diagnostic, but repeated seeds on the same benchmark item are not fully independent, so we treat these pooled p-values as descriptive rather than as a hierarchy-corrected uncertainty estimate. We report one-sided tests for the directional claim that SEGD is better than the reference method and apply Holm correction within the main-table family. For the auxiliary MC-QA rows in Table 3, we use the same pooled paired diagnostic; for ToolUse, only seed-level aggregate outputs were saved in the noise-sweep summary, so we report their means and standard deviations in Table 3 without a Wilcoxon test based on only three seeds.

E Qualitative Examples

Worked example: question, wrong traces, and summary. One representative SEGD win is problem `00a50b6b9d61` (Intermediate Algebra, Level 4):

> **Question.** For a sequence $A = (a_1, a_2, \ldots)$ with all second differences equal to 1, and with $a_{19} = a_{92} = 0$, find a_1.

The gold answer is $\boxed{819}$. On this example, SDFT predicts $\boxed{-171}$, whereas SEGD predicts the correct answer $\boxed{819}$.

The sampled incorrect attempts are long, but they repeatedly fall into the same family of mistakes. For example:

> **Wrong trace A (abridged).** Assume $a_n = a_1 + \frac{(n-1)n}{2}$, then use $a_{19} = 0$ to infer $a_1 = -171$.

> **Wrong trace B (abridged).** Write $a_k = a_1 + (k-1)b_1 + \frac{(k-2)(k-1)}{2}$, but then solve the constants incorrectly after substituting $a_{19} = a_{92} = 0$.

Instead of injecting a raw wrong trace into the teacher prompt, SEGD compresses the observed unique wrong traces into a single summary:

> *Generated summary (abridged).* The incorrect attempts either impose an overly restrictive sequence form that implicitly fixes the initial first difference, or make algebraic errors when solving the constraints $a_{19} = 0$ and $a_{92} = 0$, leading to an incorrect value of a_1.

This example illustrates the intended SEGD behavior: multiple noisy wrong traces are reduced to a compact description of a shared error pattern, providing the teacher with problem-aligned failure evidence while the gold solution remains available in the prompt.

Worked example: gate on. Example `42c8c7070e5a` (Geometry, Level 4) has evidence score 0.6667 and the gate is **on**. The question asks for $QR - RD$ in a rectangle geometry construction. Here the sampled attempts provide high error evidence: they lead to several wrong answers, while repeatedly circling related geometric misconceptions:

> **Wrong trace A.** Boxes $\boxed{9}$ after incorrect triangle-similarity reasoning.
> **Wrong trace B.** Derives inconsistent values for PD and QD and never reaches a consistent final expression.
> **Wrong trace C.** Introduces coordinates, but then computes lengths incorrectly and ends up with an expression like $\sqrt{65} - 4$.

The generated summary for this gate-active example is:

> *Generated summary (abridged).* The incorrect attempts repeatedly misapply the Pythagorean theorem, make incorrect assumptions about coordinates and segment lengths, and use flawed triangle-similarity and proportionality arguments, leading to inconsistent values for PD, QD, QR, and RD.

This is the regime where the error-evidence gate is intended to activate: many wrong traces, multiple distinct wrong answers, and shared geometric failure themes worth summarizing. The example illustrates how diverse error evidence can supply teacher context in this example.

Worked example: gate off. Example `eaf09363b43e` (Algebra, Level 4) has evidence score 0.0833 and the gate is **off**. The question asks for the value of $9\&2$ under a custom operation, and the correct answer is $\boxed{\dfrac{3\sqrt{3}}{4}}$. In this case, 11 of the 12 sampled attempts are already correct, and the one wrong trace is only a small simplification slip:

> **Wrong trace.** It simplifies $\sqrt{\frac{27}{16}}$ to $\frac{3}{4}$ instead of $\frac{3\sqrt{3}}{4}$.

The generated summary is correspondingly narrow:

> *Generated summary (abridged).* The primary mistake is simplifying $\sqrt{\frac{27}{16}}$ to $\frac{3}{4}$ instead of $\frac{3\sqrt{3}}{4}$, because $\sqrt{27} = 3\sqrt{3}$ rather than 3.

Here the gate correctly stays off: there is no strong error-evidence set to use, just a single isolated arithmetic slip amid mostly correct traces.

Teacher-grading sanity check. We also compared how the SDFT and SEGD teacher prompts score a fixed student trajectory token by token. The comparison fixes the student trajectory and varies the teacher context: when the gate is off, the SEGD teacher prompt collapses exactly to the SDFT prompt, and the per-token probabilities are identical on example `eaf09363b43e`. On the worked algebra example `00a50b6b9d61`, scoring the same wrong SDFT trajectory under the two prompts shows that the compact-summary SEGD teacher changes the distribution most clearly around the first reasoning divergence rather than at the final boxed answer: the mean log-prob over that divergence region shifts from -0.2698 under SDFT to -0.2401 under SEGD. In a batch audit over 28 gate-active examples, on fixed sampled wrong trajectories, the boxed wrong answer becomes *more*



Figure 3: Full student trajectory for the gate-active geometry example `42c8c7070e5a`, scored under the SDFT and SEGD teacher prompts. The figure also shows the original question, the injected summary, the correct gold answer, and two representative wrong traces. Red tokens are assigned higher probability by the compact-summary SEGD teacher; blue tokens are assigned higher probability by the SDFT teacher.

likely under SEGD in 12 of 17 cases (mean delta $+0.1217$). At the same time, the same audit shows that SEGD's own *correct* candidate answers are essentially unchanged at the final boxed span (mean delta -0.0002). We therefore use teacher grading only to check prompt conditioning. The observed prompt equality on gate-off cases and probability shifts on gate-active cases are consistent with the intended conditioning behavior. The increase on wrong answer spans does not demonstrate error correction. Because evidence can contain information from the scored trajectory, self-conditioning is a possible explanation; this audit does not isolate it. Equation (6) depends on the centered shift $h_j - \mathbb{E}_p[h]$, so a rise in a wrong token's teacher probability alone does not determine the student update or free-generation accuracy.

On this gate-active example, the correct answer is $\boxed{0}$, while representative wrong traces either box a wrong answer or fail to reach a consistent boxed answer at all. The full-trajectory mean log-prob changes from -0.3203 under SDFT to -0.3008 under SEGD. Figure 3 shows that this change is not

concentrated only at the final answer token. The strongest token-level shifts appear around the part of the wrong solution where it starts a triangle-similarity argument and then uses quantities such as PD, AD, and ratio equations to continue that argument. In other words, the compact-summary SEGD teacher is changing how it scores the *wrong line of reasoning*, not just the final wrong answer. The broader audit also shows that SEGD can raise the local probability of a wrong boxed answer on a fixed wrong trace, while leaving its own correct candidate answers essentially unchanged. Together, these observations characterize how error conditioning changes teacher grading around the student's observed mistake pattern.

F Controlled Error-Context Diagnostic

Table 6: Relative accuracy change from gold-only. Changes are computed as $100(A/A_{\text{gold}} - 1)$ using unrounded accuracies.

Teacher context	Relative change (%)
Gold-only	0.00
Raw error evidence	−1.99

Raw error evidence may not improve distillation: incorrect attempts can contain irrelevant reasoning or misleading claims that distract the teacher from useful corrective information. In our development diagnostic, raw-error conditioning yielded lower observed accuracy than gold-only conditioning, although the uncertainty interval included zero.