CONFIDENCE-GATED STEP SELECTION FOR LONG-COT SUPERVISED FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Supervised fine-tuning (SFT) on long chain-of-thought (CoT) reasoning traces is an effective way to improve reasoning models. Standard SFT applies one-hot supervision to every sampled token, even in steps where current model confidence exceeds the reference score from the trace generator. Using distribution matching as an analytical reference, we show that hard- and soft-target supervision can yield opposite gradient signs for the sampled token's logit. This observation motivates selecting supervision according to the model's evolving confidence. We propose **Confidence-Gated Step Selection (CGSS)**, a selective SFT method that dynamically masks reasoning-step losses using the gap between reference scores and current model confidence. Reference scores are precomputed for each step, while current scores reuse the training forward pass. The criterion operates on a fixed corpus of reasoning traces and requires neither token-level alignment nor full-vocabulary teacher distributions during SFT. Experiments across five benchmarks show improvements over vanilla SFT and heuristic selection baselines, achieving an average accuracy gain of up to 6.3 percentage points while reducing step-level confidence discrepancies, with no additional inference-time computation.

1 INTRODUCTION

Recent advances in large reasoning models (LRMs) have demonstrated the potential of transferring complex reasoning capabilities to smaller language models. Notably, DeepSeek (DeepSeek-AI et al., 2025) showed that fine-tuning dense models on reasoning trajectories produced by DeepSeek-R1 yields substantial performance gains. The simplicity of supervised fine-tuning (SFT), together with the deployment efficiency of smaller models, makes this approach attractive for improving reasoning capabilities. In this setting, extended chain-of-thought (CoT) traces (Wei et al., 2023) serve as the primary supervisory signal. However, standard SFT treats each sampled trace as a fixed supervision target and applies next-token prediction loss across all of its tokens. This uniform supervision does not account for differences in teacher confidence across reasoning steps or adapt the selection of supervision as the student evolves.

Long CoT trajectories contain exploratory or transitional steps with low confidence scores from the teacher, which we use as reference scores. We observe that vanilla SFT can amplify current model confidence relative to these scores. As shown in Figure 1 (red dashed box), the highlighted step has a low reference score, while the initial model assigns it higher confidence. After SFT, this discrepancy is further amplified rather than corrected. This behavior suggests that single-sample supervision can make the student over-commit to particular trajectory instances instead of preserving the teacher's distributional uncertainty. Recent studies have empirically observed that distilled models may diverge from teacher distributions by amplifying specific, sometimes detrimental behaviors (Liu et al., 2025; Yan et al., 2026). These observations motivate examining the relation between sampled supervision and local optimization signals.

To explain this behavior, we use distribution matching as an analytical reference for the optimization dynamics of SFT. A sampled teacher trajectory supplies one-hot targets, whereas the reference preserves the teacher's uncertainty over alternative continuations. For a fixed prefix, if the sampled token's student probability exceeds the teacher's, the hard- and soft-target losses have opposite derivative signs with respect to that token's logit. We call this local sign conflict *gradient misalignment*.

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.

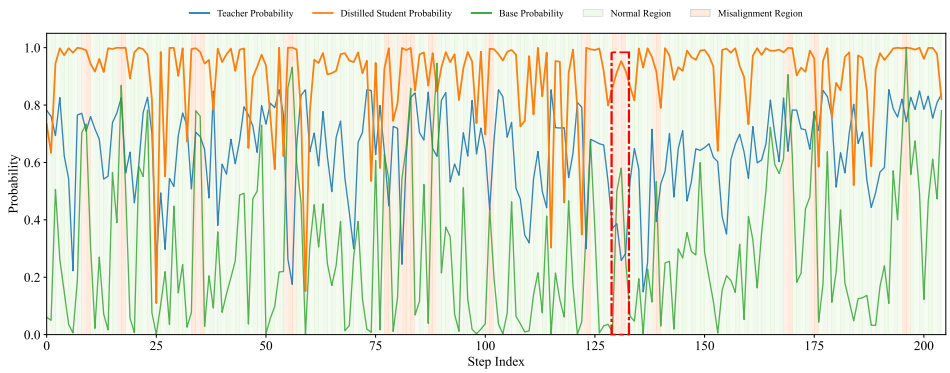


Figure 1: **Step-level confidence tracing.** The horizontal axis indexes reasoning steps; the vertical axis shows geometric-mean token-probability scores on each step, as defined in Eq. (6). The curves show the teacher (blue), initial student (green, labeled "Base Probability"), and SFT-trained student (orange). The background colors indicate the region types: green bands represent normal regions, whereas orange bands highlight misalignment regions. In the highlighted regions, the initial student has a higher confidence score than the teacher, and the SFT-trained student shows a larger gap. These traces measure confidence scores rather than gradients.

The reference helps identify potentially conflicting supervision; the method we develop remains selective SFT on a fixed corpus of reasoning traces.

Based on these observations, we propose **Confidence-Gated Step Selection (CGSS)**, a selective SFT approach that uses the step confidence gap to guide supervision. During data preparation, we compute and cache the teacher's confidence scores for individual reasoning steps. During SFT, we compare these reference scores with the model's current confidence and retain losses only for steps that satisfy the selection criterion.

To examine our theoretical analysis and evaluate CGSS, we conduct experiments across multiple long-chain reasoning benchmarks. Compared with vanilla SFT and heuristic filtering methods, CGSS improves reasoning performance, achieving an average accuracy gain of up to **6.3** percentage points. These findings support the utility of adapting step-level supervision to the current model.

Overall, the main contributions of this paper are as follows:

1. We analyze gradient misalignment in long-CoT SFT using a distribution-matching reference. The token-level analysis identifies confidence regimes in which hard- and soft-target gradients have opposite signs for the sampled token's logit.
2. We propose Confidence-Gated Step Selection (CGSS), a selective SFT method that dynamically filters reasoning-step losses using the step confidence gap. CGSS selects steps using a confidence-gap criterion without requiring token-level alignment, additional annotations, or full-vocabulary teacher distributions during SFT.
3. Experiments on multiple benchmarks show that CGSS improves over standard SFT and heuristic baselines in reasoning accuracy, supporting the role of step-level selection in long-chain reasoning SFT.

2 Problem Analysis: Empirical Observations and Gradient Dynamics

To identify the factors behind gradient misalignment in long-CoT SFT, we analyze the dynamics of SFT on teacher-generated trajectories. We introduce sequence-level distribution matching as an analytical reference and relate its single-sample form to SFT (Section 2.1). We then use *Step-level Confidence Tracing* to compare the teacher, initial student, and SFT-trained student on the same trajectory (Section 2.2; Figure 1). Finally, we identify when one-hot and soft-target supervision yield opposite gradient signs for a sampled token's logit (Section 2.3).

2.1 Distribution Matching as a Reference for SFT

Setup and notation. Let $\boldsymbol{M}_T$ be the teacher, $\boldsymbol{M}_D^{(0)}$ the initial student, and $\boldsymbol{M}_D$ the student initialized from $\boldsymbol{M}_D^{(0)}$ and updated during training. Given an input query $\boldsymbol{q}$, the teacher induces a distribution over the space of all possible output sequences $y \in \mathcal{Y}$, denoted by $P_T(y \mid \boldsymbol{q})$, where $\mathcal{Y}$ is the set of all possible responses.

Definition 2.1 (Sequence-level distribution-matching reference)**.** Following the sequence-level formulation of Kim & Rush (2016), consider aligning the student distribution $P_D(\cdot \mid \boldsymbol{q})$ with the teacher distribution $P_T(\cdot \mid \boldsymbol{q})$ by minimizing

$$\min_{\boldsymbol{\theta}} \; D_{\mathrm{KL}}\Big(P_T(\cdot \mid \boldsymbol{q}) \;\Big\|\; P_D(\cdot \mid \boldsymbol{q}; \boldsymbol{\theta})\Big), \tag{1}$$

where $\boldsymbol{\theta}$ denotes the trainable student parameters. This objective supplies a reference direction for analysis; CGSS optimizes the masked SFT loss in Eq. (13).

Expanding Eq. (1) gives

$$D_{\mathrm{KL}}(P_T \| P_D) = \sum_{y \in \mathcal{Y}} P_T(y \mid \boldsymbol{q}) \, \log \frac{P_T(y \mid \boldsymbol{q})}{P_D(y \mid \boldsymbol{q}; \boldsymbol{\theta})}. \tag{2}$$

Dropping the teacher-only constant term yields the equivalent objective:

$$\mathcal{L}_{\mathrm{SEQ}}(\boldsymbol{\theta}) = -\mathbb{E}_{y \sim P_T(\cdot \mid \boldsymbol{q})}\big[\log P_D(y \mid \boldsymbol{q}; \boldsymbol{\theta})\big]. \tag{3}$$

This reference is related to classical soft-target distillation (Hinton et al., 2015); here it is used to analyze SFT updates on teacher-generated data.

Single-sample supervision. Since the output space $\mathcal{Y}$ is exponentially large, the expectation in Eq. (3) is computationally intractable. The standard approach approximates this expectation using a single Monte Carlo sample $\hat{y}$ from the teacher distribution $P_T(\cdot \mid \boldsymbol{q})$. This implicitly replaces the full teacher distribution with a point mass at $\hat{y}$:

$$P_T(y \mid \boldsymbol{q}) \approx \mathbf{1}\{y = \hat{y}\}. \tag{4}$$

Under this approximation, the sequence-level objective reduces to the standard SFT objective:

$$\mathcal{L}_{\mathrm{SFT}}(\boldsymbol{\theta}) = -\log P_D(\hat{y} \mid \boldsymbol{q}; \boldsymbol{\theta}). \tag{5}$$

Eq. (5) is our SFT baseline on a fixed corpus of teacher trajectories. CGSS changes which step losses contribute to this objective.

2.2 Step-level Confidence Tracing

Section 2.1 relates sampled SFT supervision to a distribution-matching reference. We now examine how this supervision affects confidence along a reasoning trajectory.

We examine changes in the student's confidence profile, motivated by preserving the teacher's calibrated uncertainty. Our *Step-level Confidence Tracing* measures scalar scores on observed steps; it does not measure calibration error or full-distribution agreement.

We choose this step-wise granularity because token-level tracing is often difficult to compare across models with different tokenizers. We score the same step text using the geometric mean of token probabilities under each model's tokenizer. These are length-normalized scores, not normalized probabilities over possible steps.

Sentence-level segmentation. Given a teacher reasoning trajectory y, we segment it into K consecutive steps $y = (s_1, s_2, \ldots, s_K)$, where each s_k is a sentence-like span. To obtain consistent segmentation across natural language and mathematical expressions, we split sentences using a heuristic pattern: terminal punctuation followed by whitespace and an uppercase letter. [1] This segmentation procedure covers the most common sentence structures and encourages steps to correspond to semantically complete units.

[1] Specifically, we implement this using the Python regular expression `r'([.?!}\]])([\s\n]+)([A-Z])'`.

Step-level confidence score. For the k-th step s_k in the reasoning trajectory, let it be represented as a sequence of tokens $(t_1, \ldots, t_N)$ under model $\boldsymbol{M}$'s tokenizer. We define the step-level confidence score as the geometric mean of the token probabilities, which serves as a length-normalized confidence measure for comparing teacher and student behavior on the same semantic step:

$$\mathrm{Prob}(s_k; \boldsymbol{M}) \triangleq \exp\left(\frac{1}{N}\sum_{i=1}^{N} \log P_{\boldsymbol{M}}(t_i \mid \boldsymbol{q}, s_{<k}, t_{<i})\right), \tag{6}$$

where $s_{<k}$ denotes the preceding steps.

The resulting confidence-score traces are visualized in Figure 1. Specifically, we select a representative reasoning trajectory from the AceReason-1.1-SFT dataset (Chen et al., 2025). To analyze the misalignment, we track step-confidence scores under the same context across three models: the teacher model $\boldsymbol{M}_T$ (DeepSeek-R1-0528 (DeepSeek-AI et al., 2025)), the initial student model $\boldsymbol{M}_D^{(0)}$ (Qwen3-4B-Instruct), and the SFT-trained student model $\boldsymbol{M}_D$, which is trained on trajectories sampled from the teacher. This representative trace illustrates the phenomenon, while Appendix D provides a dataset-level confidence-discrepancy analysis. By tracing these trajectories of $\boldsymbol{M}_T$, $\boldsymbol{M}_D$, and $\boldsymbol{M}_D^{(0)}$, we observe increasing confidence gaps:

⋆ **Observation.** We identify reasoning steps where the teacher model assigns a relatively low confidence score. **In regions where the initial student model already has a higher confidence score than the teacher, the SFT-trained student does not reduce this deviation.** Instead, the SFT-trained student has an even higher confidence score. This indicates that SFT can move the student's confidence away from the teacher's in such regions, increasing the confidence gap rather than reducing it.

Intuitively, these steps may correspond to branching points where multiple continuations are plausible under the teacher distribution. The higher confidence scores of the SFT-trained model in these regions suggest that it over-commits to the single observed trajectory, a phenomenon we qualitatively analyze in Appendix C (see Figure 4). Rather than preserving the teacher's uncertainty over alternative continuations, the student treats the sampled path as a deterministic target.

2.3 Theoretical Analysis: Local Gradient Misalignment

The preceding observations concern step-confidence gaps. To motivate a selection rule, we distinguish the expectation of a teacher-sampled SFT gradient from the derivative on a particular sampled token. The analysis below concerns the latter's logit coordinate.

2.3.1 Expected SFT Gradient under Teacher Sampling

The sequence-level reference minimizes cross-entropy with respect to the teacher distribution P_T. SFT approximates this by minimizing $-\log P_D(\hat{y})$ on a single sample $\hat{y} \sim P_T$. Under the sampling assumption $\hat{y} \sim P_T$, the unfiltered SFT gradient is an *unbiased estimator* of the sequence-level reference gradient:

$$\mathbb{E}_{\hat{y}\sim P_T}\left[-\nabla_{\boldsymbol{\theta}} \log P_D(\hat{y} \mid \boldsymbol{q})\right] = \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathrm{SEQ}}. \tag{7}$$

This expectation is over teacher samples; it does not imply that every update on a fixed corpus, or the CGSS masked gradient, is unbiased for the reference. The space of possible reasoning paths $\mathcal{Y}$ is combinatorially large. Consequently, the training process operates in a sparse-data regime: for a specific query and prefix, the model typically observes only a small number of teacher-sampled realizations, and often only one. The expectation in Eq. 7 is therefore approximated by sparse single-sample updates in practical training. A particular sampled token can nevertheless yield a logit derivative with the opposite sign from the soft-target reference. We characterize this local sign conflict below. This issue is particularly relevant in long-CoT SFT, where a single trajectory contains many intermediate reasoning steps at which such single-sample updates may be applied. Thus, the local analysis below is not specific to long-CoT, but long-CoT provides a setting in which such local directional conflicts can appear repeatedly along the training signal.

2.3.2 Gradient Misalignment: Distribution Matching vs. Point-Mass Fitting

To analyze a possible source of the confidence gaps, we examine a sampled-token logit derivative. Although the sequence-level reference (Eq. 1) is defined over full trajectories, autoregressive de-

composition relates sequence-level KL to an expected sum of token-level soft cross-entropies (see Appendix B for derivation). This allows us to isolate the optimization conflict by analyzing the gradient for a single token.

Consider the gradient with respect to the logit of the sampled token $v^\star$. The following comparison follows directly from the standard gradients of soft-target and hard-label cross-entropy; the key point is its implication in the sparse single-sample regime. Comparing the soft-target reference $\mathcal{L}_{\mathrm{REF}}$ against the SFT loss $\mathcal{L}_{\mathrm{SFT}}$ reveals a target-alignment mismatch.

Proposition 2.2 (Gradient Discrepancy)**.** *At a fixed prefix over a common token space, let $q(v^\star)$ and $p(v^\star)$ be teacher and student probabilities of sampled token $v^\star$. For soft-target and hard-label cross-entropy, its logit derivatives are:*

$$\nabla_{\boldsymbol{z}(v^\star)}\mathcal{L}_{\mathrm{REF}} = \underbrace{p(v^\star) - q(v^\star)}_{\textit{Calibrate to Teacher}}, \tag{8}$$

$$\nabla_{\boldsymbol{z}(v^\star)}\mathcal{L}_{\mathrm{SFT}} = \underbrace{p(v^\star) - 1}_{\textit{Force to Certainty}}. \tag{9}$$

Mechanism: Directional divergence. Eq. (8) drives the student to align with the teacher's probability $q(v^\star)$, thereby preserving the soft distributional target specified by the teacher. In contrast, due to one-hot supervision, Eq. (9) drives $p(v^\star)$ toward 1. This update treats the sampled token as a point-mass target and does not account for alternative continuations that may receive non-negligible probability under the teacher.

Figure 2 illustrates opposite update signs for this logit when $q(v^\star) < p(v^\star) < 1$. Consider a scenario where the student is already more confident than the teacher (e.g., $q(v^\star) = 0.3$ and $p(v^\star) = 0.5$).

- **Soft-target reference ($\nabla > 0$):** Since $p > q$, the gradient is positive, pushing the logit down to reduce confidence and align with the teacher.
- **SFT ($\nabla < 0$):** Since $p < 1$, the gradient remains negative, pushing the logit up to further amplify confidence.

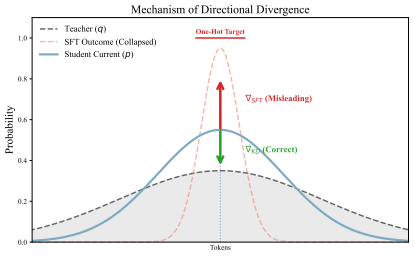


Figure 2: **Directional divergence.** When $q < p < 1$, the reference (KD) lowers the sampled-token logit; SFT raises it.

In this regime ($q < p < 1$), gradient descent on the sampled-token logit raises it under SFT and lowers it under the soft-target reference. We refer to this coordinate-level sign conflict as **local gradient misalignment**.

⋆ **Insight. When $q(v^\star) < p(v^\star) < 1$, hard- and soft-target losses have opposite derivative signs for the sampled token's logit.** This local result motivates a confidence-gap selection signal. It does not establish opposing full parameter-gradient vectors, certify step correctness, or guarantee gradient agreement for every token in a selected step.

3 CGSS: Confidence-Gated Step Selection

Building on Section 2, we propose **Confidence-Gated Step Selection (CGSS)**, illustrated in Figure 3. CGSS uses a dynamic confidence-gap criterion to select step losses in supervised fine-tuning on a fixed corpus of reasoning traces.

Our focus is supervision selection within SFT on a fixed corpus of reasoning traces. CGSS retains hard-label next-token prediction and uses precomputed reference scores only to determine which step losses contribute to training. We therefore compare selection strategies under the same data and training protocol.

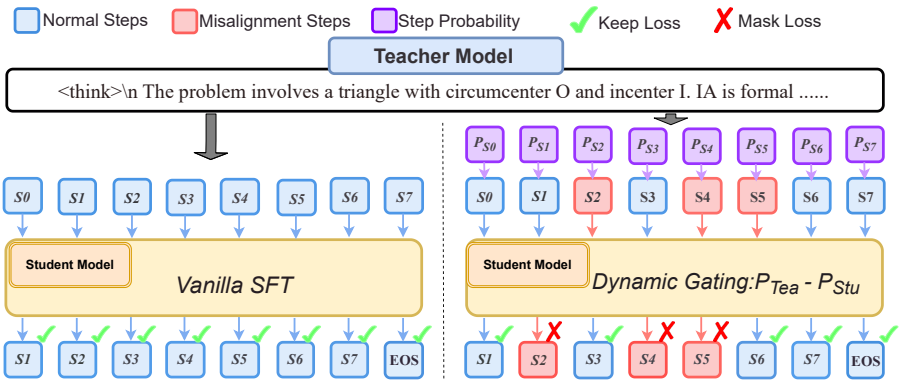


Figure 3: **Overview of Confidence-Gated Step Selection (CGSS).** The framework operates in three stages: (1) **Trajectory Generation**: The teacher produces reasoning chains, which are segmented into semantic steps while computing step-confidence scores. (2) **Dynamic Gating**: We calculate the confidence gap (Δ) between the teacher and the student. Losses for steps that fail the criterion are masked out. (3) **Selective Optimization**: Retained step losses update the model; the complete trajectory remains in the context.

3.1 Step-Level Confidence Score

For step selection, we use the **geometric-mean confidence score** defined in Section 2.2. This length-normalized score uses each model's own tokenizer and requires no token-level alignment.

For a reasoning step s_k consisting of N tokens $t_{1:N}$ under the teacher's tokenizer, the teacher's confidence score is calculated as:

$$\mathrm{Prob}_T(s_k) = \exp\left(\frac{1}{N}\sum_{i=1}^{N}\log P_T(t_i \mid \boldsymbol{q}, s_{<k}, t_{<i})\right). \tag{10}$$

Correspondingly, the student's confidence score $\mathrm{Prob}_D(s_k)$ is computed over the same step text under the student's tokenizer. Their difference is a step-level proxy motivated by the local analysis. Different tokenizations can affect the scores; the proxy does not guarantee gradient agreement for every token.

3.2 Dynamic Gating Mechanism

With this confidence metric, we construct a criterion for potentially conflicting updates. Our **dynamic selection mechanism** uses the confidence gap as a practical signal motivated by Section 2.3.

The process operates in a decoupled manner for computational efficiency. ***Firstly***, during data generation, we compute the teacher confidence score (Prob_T) from sampled-token probabilities and cache it as the reference score for each step. ***Secondly***, during SFT, we compute current step-confidence scores (Prob_D) dynamically. This computation reuses the logits produced by the standard forward pass for loss calculation, so acquiring Prob_D does not require an additional teacher forward pass or full-vocabulary teacher distribution during SFT. This design allows the filtering decision to reflect the model's current learning state while keeping the training procedure close to standard SFT.

Formally, we define the *confidence gap* for step s_k as:

$$\Delta(s_k) = \mathrm{Prob}_T(s_k) - \mathrm{Prob}_D(s_k). \tag{11}$$

A positive $\Delta(s_k)$ indicates that the teacher assigns higher confidence than the student to the same reasoning step.

We apply a margin-based gating function to generate a binary mask M_k:

$$M_k = \mathbf{1}\big(\Delta(s_k) > \gamma\big), \tag{12}$$

where γ is a hyperparameter controlling the strictness of the selection.

3.3 SELECTIVE SFT OBJECTIVE

Finally, we define the optimization objective at the step level. Unlike standard SFT, which maximizes the likelihood of every token in the sampled trajectory, our method applies the loss only to steps satisfying the confidence-gap criterion. The loss function is formulated as:

$$\mathcal{L}_{\mathrm{CGSS}}(\boldsymbol{\theta}) = -\frac{1}{\sum_{k=1}^{K} M_k \cdot |s_k|} \sum_{k=1}^{K} M_k \cdot \log P_D(s_k \mid \boldsymbol{q}, s_{<k}; \boldsymbol{\theta}). \tag{13}$$

where $|s_k|$ denotes the student-token length of step s_k. Masking changes the loss contribution; all steps remain in the teacher-forced context. By optimizing this masked SFT objective, we aim to reduce potentially conflicting updates in the single-sample regime. We evaluate this criterion through reasoning accuracy and step-level teacher-student confidence discrepancy.

4 EXPERIMENTS

We evaluate step-selection strategies for SFT on a fixed corpus of reasoning traces, using the same models, training data, and evaluation protocol across methods.

4.1 EXPERIMENTAL SETUP

Models. We evaluate CGSS and comparison methods on Qwen3-4B-Base, Qwen3-8B-Base, Qwen3-4B-Instruct-2507, and Qwen2.5-7B-Instruct (Yang et al., 2025) since they represent diverse parameter scales and alignment types that are widely used in reasoning tasks. All experiments are run on an 8xNVIDIA A100 GPU server.

Datasets. We evaluate on five reasoning benchmarks: AIME24, AIME25 (AIME, 2025), MATH500 (Hendrycks et al., 2021), OlympiadBench (He et al., 2024), and GPQA-Diamond (Rein et al., 2023), covering competition mathematics and graduate-level scientific reasoning. Detailed benchmark descriptions are provided in Appendix E.

Training Data. For training data construction, we randomly sample 10,000 problems from the AceReason-1.1-SFT dataset (Chen et al., 2025). For these problems, we use DeepSeek-R1-0528 (DeepSeek-AI et al., 2025) to generate high-quality long CoT reasoning trajectories. These data constitute our supervised finetuning dataset.

Baseline. We compare CGSS with the following baseline methods: ***Vanilla SFT***: Standard supervised finetuning that trains on all steps in the CoT, serving as the primary reference baseline. ***TDST*** (Cui et al., 2025b): Target-Distribution Supervision Tailoring (TDST) prioritizes steps with higher length-normalized likelihood (equivalently, lower perplexity) under the student model. ***EDSP*** (Li et al., 2025): Entropy-Driven Supervision Prioritization (EDSP) that prioritizes training signal from CoT steps with higher predictive entropy. Concretely, it performs high-entropy selection, emphasizing high-entropy reasoning steps during training.

Evaluation Setup. For consistent evaluation, all evaluations are conducted under unified settings. Specifically, we set the sampling temperature to 0.6 and Top-p to 0.95. For each test problem, we sample 4 responses with a maximum generation length of 32,768 tokens. We report the average accuracy on each benchmark.

4.2 RESULTS AND DISCUSSION

Table 1 compares models on the evaluated benchmarks. Up-arrow annotations in the result tables denote percentage-point gains over the same model's Vanilla SFT result. We mainly compare Vanilla SFT with our CGSS method. The results show that CGSS improves over Vanilla SFT on most tasks while restricting loss contributions to selected step tokens. For a comprehensive breakdown of experimental results, including comparisons with additional heuristic baselines, please refer to Appendix G.

Table 1: **Main Results.** We report the accuracy (%) on math and reasoning benchmarks. **Bold** indicates the best result.

Model	Method	AIME24	AIME25	MATH500	OlympiadBench	GPQA	Avg.
Qwen2.5 7B-Instruct	+ Vanilla SFT	24.20	21.60	79.80	44.20	33.10	40.60
	+ EDSP	15.00	16.70	80.30	44.10	31.30	37.50
	+ TDST	16.70	23.30	81.90	43.00	34.10	39.80
	+ CGSS (Ours)	**27.50**$_{\uparrow 3.3}$	**25.00**$_{\uparrow 3.4}$	**83.10**$_{\uparrow 3.3}$	**45.20**$_{\uparrow 1.0}$	**34.70**$_{\uparrow 1.6}$	**43.10**$_{\uparrow 2.5}$
Qwen3 4B-Base	+ Vanilla SFT	35.00	26.70	87.00	50.70	43.80	48.60
	+ EDSP	31.70	28.30	85.90	51.30	45.50	48.50
	+ TDST	37.50	26.70	86.00	50.60	47.50	49.70
	+ CGSS (Ours)	**42.50**$_{\uparrow 7.5}$	**31.70**$_{\uparrow 5.0}$	**88.20**$_{\uparrow 1.2}$	**52.70**$_{\uparrow 2.0}$	**47.90**$_{\uparrow 4.1}$	**52.60**$_{\uparrow 4.0}$
Qwen3 8B-Base	+ Vanilla SFT	45.00	29.20	88.20	54.70	52.90	54.00
	+ EDSP	44.20	31.70	89.40	54.80	54.90	55.00
	+ TDST	45.00	32.50	89.00	54.00	53.80	54.90
	+ CGSS (Ours)	**50.00**$_{\uparrow 5.0}$	**36.70**$_{\uparrow 7.5}$	**89.60**$_{\uparrow 1.4}$	**55.70**$_{\uparrow 1.0}$	**56.90**$_{\uparrow 4.0}$	**57.80**$_{\uparrow 3.8}$
Qwen3 4B-Instruct	+ Vanilla SFT	51.70	37.50	83.90	54.40	51.10	55.70
	+ EDSP	52.50	43.30	88.60	57.60	53.20	59.00
	+ TDST	49.20	43.30	85.40	57.10	53.70	57.70
	+ CGSS (Ours)	**58.30**$_{\uparrow 6.6}$	**45.00**$_{\uparrow 7.5}$	**91.40**$_{\uparrow 7.5}$	**59.80**$_{\uparrow 5.4}$	**55.40**$_{\uparrow 4.3}$	**62.00**$_{\uparrow 6.3}$

CGSS improves average accuracy across all evaluated models. On Base models, Qwen3-4B-Base improves from 48.6% to 52.6%, and Qwen3-8B-Base from 54.0% to 57.8%, gains of 4.0 and 3.8 percentage points. On Instruct models, Qwen3-4B-Instruct improves from 55.7% to 62.0%, and Qwen2.5-7B-Instruct from 40.6% to 43.1%, gains of 6.3 and 2.5 percentage points. These results support the selection criterion across the evaluated model families and scales.

Several task-level gains are substantial. On AIME24, Qwen3-4B-Base improves by 7.5 percentage points (from 35.0% to 42.5%) and Qwen3-4B-Instruct by 6.6 points (from 51.7% to 58.3%). On AIME25, Qwen3-8B-Base gains 7.5 points (from 29.2% to 36.7%). Qwen3-4B-Instruct also improves on MATH500, from 83.9% to 91.4%, a gain of 7.5 points.

Discussion. These results support using the confidence-gap criterion to select supervision. Accuracy gains alone do not directly measure gradient agreement. Additional analysis in Appendix D shows smaller step-level confidence discrepancies under CGSS.

4.3 Ablation Study

To examine the contributions of confidence-based selection and dynamic scoring, we conduct ablation studies on the Qwen3-4B-Base model comparing CGSS with two variants: (1) **Random Selection**: We randomly mask steps at a global retention rate matching the average of CGSS over the full training run, controlling for average step retention. (2) **Static Selection**: Instead of computing the model's confidence score dynamically, we pre-compute it offline using the frozen initial model before training starts.

Table 2 summarizes the results. First, Random Selection lowers average accuracy from 48.6% to 47.1%. At a matched average step-retention rate, CGSS performs better, supporting the role of the selection criterion beyond the number of retained steps. Second, while Static Selection outperforms Vanilla SFT (49.8% vs 48.6%), it still lags behind the full dynamic CGSS (52.6%). This supports updating selection with the current model: as training progresses, the model's confidence profile evolves. A step retained early in training may fail the confidence-gap criterion later. Static selection fails to capture this shift, whereas CGSS dynamically updates the mask using the current confidence gap, aiming to reduce potentially conflicting updates.

Table 2: **Ablation study of different selection strategies on Qwen3-4B-Base.** We report accuracy (%) across all five benchmarks.

Strategy	AIME24	AIME25	MATH500	OLY.	GPQA	Avg.
Vanilla SFT	35.0	26.7	87.0	50.7	43.8	48.6
Random Selection	33.3	25.0	86.2	49.2	41.9	47.1
Static Selection	37.5	27.8	87.1	51.0	45.4	49.8
CGSS (Ours)	**42.5**	**31.7**	**88.2**	**52.7**	**47.9**	**52.6**

5 Related Work

CoT prompting and fine-tuning use intermediate reasoning traces to support problem solving (Wei et al., 2022; Zelikman et al., 2022; DeepSeek-AI et al., 2025). Long traces also increase the amount of computation required for training. Work on reasoning efficiency explores token budgets (Han et al., 2024), continuous latent representations (Hao et al., 2024), and the effects of trace length (Wu et al., 2025). Perplexity-guided refinement (Cui et al., 2025a) and step-entropy compression (Li et al., 2025) further identify parts of a trace that can be shortened. These directions concern the content or representation of reasoning. CGSS keeps the complete trace in the conditioning context and selects which steps contribute to the training loss.

Selective objectives study how different parts of a sequence contribute to learning. Gradient attribution offers a complementary view of how input features contribute to model predictions (Sundararajan et al., 2017). Goldfish loss excludes randomly selected token losses to reduce memorization (Hans et al., 2024), while Rho-1 uses reference scores to focus language-model training on selected tokens (Lin et al., 2024). Related studies examine data suitability for instruction tuning (Zhang et al., 2025a) and the role of high-entropy tokens in reinforcement learning (Wang et al., 2025). Together, these approaches motivate examining the allocation of training signals at different granularities. Our selection unit is a reasoning step, and its loss mask is recomputed using the model's current confidence during SFT.

Recent work also studies distribution alignment and adaptive supervision at sequence and token granularity (Yan et al., 2026; Huang et al., 2025; Zhang et al., 2025b). These studies examine both the information supplied as supervision and how it is used during learning. CGSS focuses on how the confidence gap of a reasoning step changes during training on a fixed corpus. The gap combines a precomputed score with current model confidence and serves as a selection signal for hard-label supervision. This connects the local gradient analysis to a practical choice of step losses, while keeping the optimization procedure close to standard SFT. The criterion is a step-level proxy: its local token-level motivation does not imply a guarantee for every token.

6 Conclusion

We studied supervision selection for long-CoT SFT using distribution matching as an analytical reference. The token-level analysis identifies confidence regimes with opposite hard- and soft-target gradient signs for the sampled token's logit. Motivated by this observation, CGSS selects reasoning-step losses using the evolving confidence gap. Step scores are cached during data construction, and current model scores reuse the standard training forward pass. All steps remain in the conditioning context, so the method changes where supervision is applied while preserving the original reasoning traces.

Experiments across five reasoning benchmarks show improved accuracy over vanilla SFT and heuristic selection baselines, together with smaller step-level confidence discrepancies. The random and static selection ablations further support both the confidence-based criterion and updating the selection mask during training. These findings suggest that the choice of supervision should account for the model's evolving state, even when the training corpus remains fixed. CGSS implements this choice through step losses and retains the standard inference procedure.

REFERENCES

AIME. AIME problems and solutions, 2025. URL https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions.

Yang Chen, Zhuolin Yang, Zihan Liu, Chankyu Lee, Peng Xu, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Acereason-nemotron: Advancing math and code reasoning through reinforcement learning. *CoRR*, abs/2505.16400, 2025.

Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, Yue Xing, Jiliang Tang, and Qi He. Stepwise perplexity-guided refinement for efficient chain-of-thought reasoning in large language models, 2025a. URL https://arxiv.org/abs/2502.13260.

Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, Yue Xing, Jiliang Tang, and Qi He. Stepwise perplexity-guided refinement for efficient chain-of-thought reasoning in large language models, 2025b. URL https://arxiv.org/abs/2502.13260.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, et al. Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning, 2024.

Abhimanyu Hans, Yuxin Wen, Neel Jain, John Kirchenbauer, Hamid Kazemi, Prajwal Singhania, Siddharth Singh, Gowthami Somepalli, Jonas Geiping, Abhinav Bhatele, and Tom Goldstein. Be like a goldfish, don't memorize! mitigating memorization in generative LLMs, 2024.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space, 2024.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024. URL https://arxiv.org/abs/2402.14008.

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. *arXiv preprint arXiv:2103.03874*, 2021.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

Haiduo Huang, Jiangcheng Song, Yadong Zhang, and Pengju Ren. SelecTKD: Selective token-weighted knowledge distillation for LLMs, 2025. URL https://arxiv.org/abs/2510.24021.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Conference on Empirical Methods in Natural Language Processing*, pp. 1317–1327, 2016.

Zeju Li, Jianyuan Zhong, Ziyang Zheng, Xiangyu Wen, Zhijian Xu, Yingying Cheng, Fan Zhang, and Qiang Xu. Compressing chain-of-thought in llms via step entropy, 2025. URL https://arxiv.org/abs/2508.03346.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Rho-1: Not all tokens are what you need, 2024.

Kaiyuan Liu, Shaotian Yan, Rui Miao, Bing Wang, Chen Shen, Jun Zhang, and Jieping Ye. Where did this sentence come from? tracing provenance in llm reasoning distillation. *arXiv preprint arXiv:2512.20908*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. *CoRR*, abs/2311.12022, 2023.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning, 2025. URL https://arxiv.org/abs/2506.01939.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Yuyang Wu, Yifei Wang, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs, 2025.

Shaotian Yan, Kaiyuan Liu, Chen Shen, Bing Wang, Sinan Fan, Jun Zhang, Yue Wu, Zheng Wang, and Jieping Ye. Distribution-aligned sequence distillation for superior long-cot reasoning, 2026. URL https://arxiv.org/abs/2601.09088.

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. *CoRR*, abs/2505.09388, 2025.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning, 2022.

Dylan Zhang, Qirun Dai, and Hao Peng. The best instruction-tuning data are those that fit. *CoRR*, abs/2502.04194, 2025a.

Songming Zhang, Xue Zhang, Tong Zhang, Bojie Hu, Yufeng Chen, and Jinan Xu. AlignDistil: Token-level language model alignment as adaptive policy distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19791–19807, Vienna, Austria, 2025b. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.972. URL https://aclanthology.org/2025.acl-long.972/.

A SCOPE: SELECTIVE SFT AND DISTRIBUTION-MATCHING REFERENCES

CGSS selects step losses within SFT on a fixed corpus of teacher-generated trajectories. The distribution-matching reference explains a local gradient conflict and motivates the confidence-gap criterion; the optimized loss remains masked hard-label cross-entropy. Teacher confidence scores are computed once during data construction, and current student scores reuse the ordinary training forward pass. Thus, the method uses scalar confidence scores to decide where to apply supervision. Although teacher-generated SFT is commonly studied within reasoning distillation, our contribution and comparisons concern supervision selection under this fixed-corpus protocol. On-policy training changes the trajectory distribution, while full-distribution objectives change the supervision supplied during optimization. The present experiments do not establish relative performance under those alternative protocols.

B EQUIVALENCE BETWEEN SEQUENCE-LEVEL KL AND TOKEN-LEVEL SOFT CE.

In this section, we provide the detailed proof for Lemma B.1 regarding the decomposition of sequence-level KL divergence.

Lemma B.1 (Chain-rule decomposition of the sequence-level reference)**.** *Let the teacher and student distributions admit the autoregressive factorization* $P_T(y \mid \boldsymbol{q}) = \prod_{t=1}^{|y|} q_t(y_t \mid \boldsymbol{q}, y_{<t})$ *and* $P_D(y \mid \boldsymbol{q}; \boldsymbol{\theta}) = \prod_{t=1}^{|y|} p_t(y_t \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})$*, respectively. The sequence-level KL divergence decomposes as:*

$$D_{\mathrm{KL}}\Big(P_T(\cdot \mid \boldsymbol{q}) \,\Big\|\, P_D(\cdot \mid \boldsymbol{q}; \boldsymbol{\theta})\Big) = \mathbb{E}_{y \sim P_T(\cdot|\boldsymbol{q})}\left[\sum_{t=1}^{|y|} D_{\mathrm{KL}}\Big(q_t(\cdot \mid \boldsymbol{q}, y_{<t}) \,\Big\|\, p_t(\cdot \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})\Big)\right]. \tag{14}$$

Proof. By definition of the Kullback-Leibler divergence, we have:

$$D_{\mathrm{KL}}(P_T \| P_D) = \mathbb{E}_{y \sim P_T}\left[\log \frac{P_T(y \mid \boldsymbol{q})}{P_D(y \mid \boldsymbol{q}; \boldsymbol{\theta})}\right] \tag{15}$$

$$= \mathbb{E}_{y \sim P_T}\left[\sum_{t=1}^{|y|} \left(\log q_t(y_t \mid \boldsymbol{q}, y_{<t}) - \log p_t(y_t \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})\right)\right]. \tag{16}$$

Using the linearity of expectation, we can move the summation outside. Furthermore, for each time step t, the expectation over the full sequence y can be rewritten using the law of total expectation, conditioning on the prefix $y_{<t}$:

$$D_{\mathrm{KL}}(P_T \| P_D) = \sum_{t=1}^{|y|} \mathbb{E}_{y \sim P_T}\left[\log \frac{q_t(y_t \mid \boldsymbol{q}, y_{<t})}{p_t(y_t \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})}\right] \tag{17}$$

$$= \sum_{t=1}^{|y|} \mathbb{E}_{y_{<t} \sim P_T}\left[\mathbb{E}_{y_t \sim q_t(\cdot|\boldsymbol{q}, y_{<t})}\left[\log \frac{q_t(y_t \mid \boldsymbol{q}, y_{<t})}{p_t(y_t \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})}\right]\right] \tag{18}$$

$$= \mathbb{E}_{y \sim P_T}\left[\sum_{t=1}^{|y|} D_{\mathrm{KL}}\Big(q_t(\cdot \mid \boldsymbol{q}, y_{<t}) \,\Big\|\, p_t(\cdot \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})\Big)\right]. \tag{19}$$

This completes the proof. □

Remark B.2 (Connection to Soft Cross-Entropy). Expanding the KL term in Eq. (14), we obtain:

$$D_{\mathrm{KL}}(q_t \| p_t) = -H(q_t) + H(q_t, p_t). \tag{20}$$

Since the entropy of the teacher distribution $H(q_t)$ is constant with respect to the student parameters $\boldsymbol{\theta}$, minimizing the sequence-level KL is equivalent to minimizing the expected sum of token-level cross-entropies:

$$\mathcal{L}_{\mathrm{REF}}(\boldsymbol{\theta}) = \mathbb{E}_{y \sim P_T}\left[\sum_{t=1}^{|y|} H\Big(q_t(\cdot \mid \boldsymbol{q}, y_{<t}),\, p_t(\cdot \mid \boldsymbol{q}, y_{<t}; \boldsymbol{\theta})\Big)\right]. \tag{21}$$

The reference retains soft teacher targets. SFT instead uses a point-mass target at the sampled token, while CGSS selects which of these hard-label losses contribute to training.

C Qualitative Analysis

We visualize reasoning steps with different teacher-student confidence relationships to illustrate the motivation for selection. Figure 4 illustrates examples from the AceReason-1.1-SFT dataset (Chen et al., 2025). The orange regions highlight steps associated with higher student confidence relative to the teacher; green marks the contrasting regions. These labels visualize scalar confidence relationships. They do not establish the semantic correctness of a step, full-distribution alignment, or the direction of its parameter-gradient vector.

Specifically, as illustrated in the Figure 4, the sentences within the orange regions represent exploratory and tentative reasoning. In these instances, the teacher model assigns lower confidence, suggesting that these moments may act as branching points with multiple plausible continuations. Conversely, the student model exhibits higher relative confidence at these junctures. Such patterns may contribute to later reasoning errors if repeatedly reinforced during training.

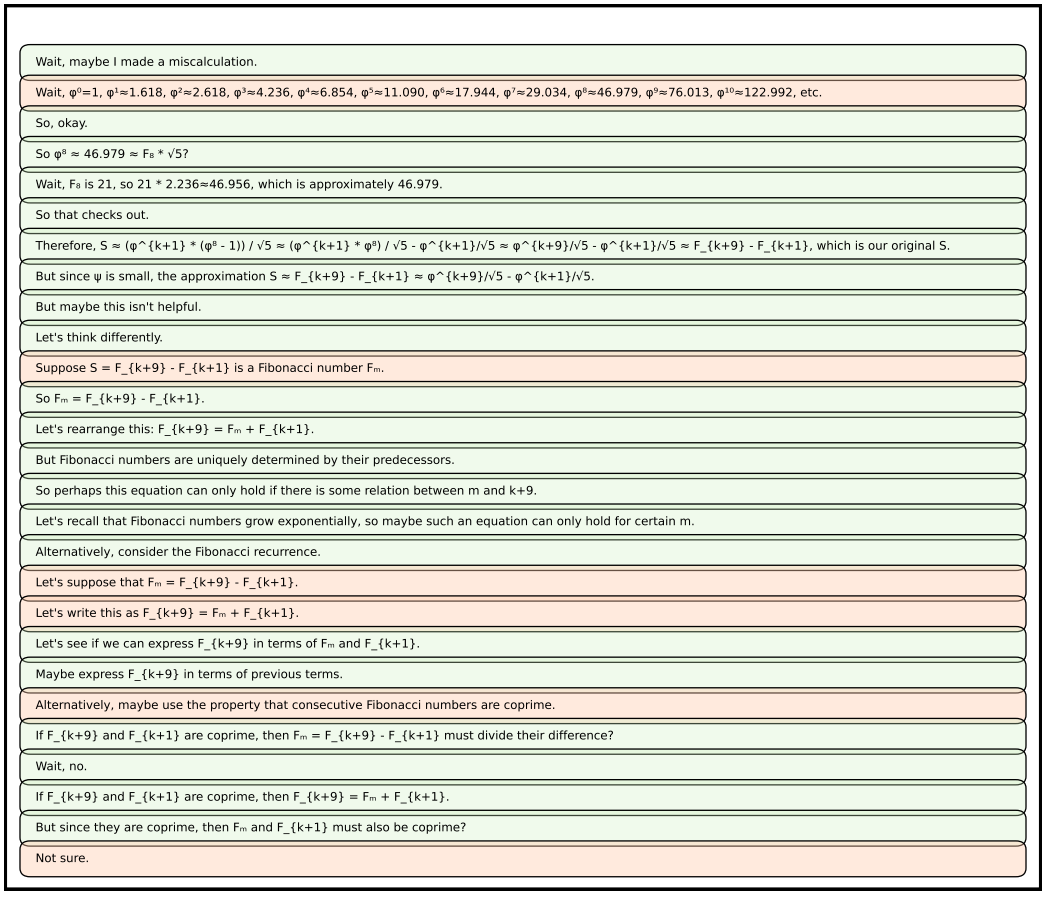


Figure 4: **Qualitative confidence-gap examples.** Case study on AceReason-1.1-SFT. Orange highlights steps associated with higher student confidence relative to the teacher; green marks contrasting regions. The colors do not label semantic correctness or measured gradient directions.

D Step-Level Confidence Discrepancy Analysis

We evaluate the step-confidence discrepancies associated with the selection signal motivated in Section 2.3. On the held-out validation set, we calculate the **Step-Level Confidence Discrepancy**:

the absolute difference between teacher and student geometric-mean confidence scores defined in Eq. (6). The horizontal axis in Figure 5 labels this quantity "Probability Discrepancy".

Figure 5 visualizes the density distribution of this discrepancy. The Vanilla SFT baseline (pink bars) exhibits a broad distribution with substantial probability mass shifted towards higher discrepancy values. This indicates larger teacher-student confidence gaps on the evaluated steps. By comparison, CGSS (blue bars) shows a more concentrated distribution with lower discrepancy values. These observations are consistent with the intended effect of the confidence-gap criterion: CGSS is associated with smaller step-level teacher-student confidence discrepancies. This measurement summarizes scalar confidence gaps.

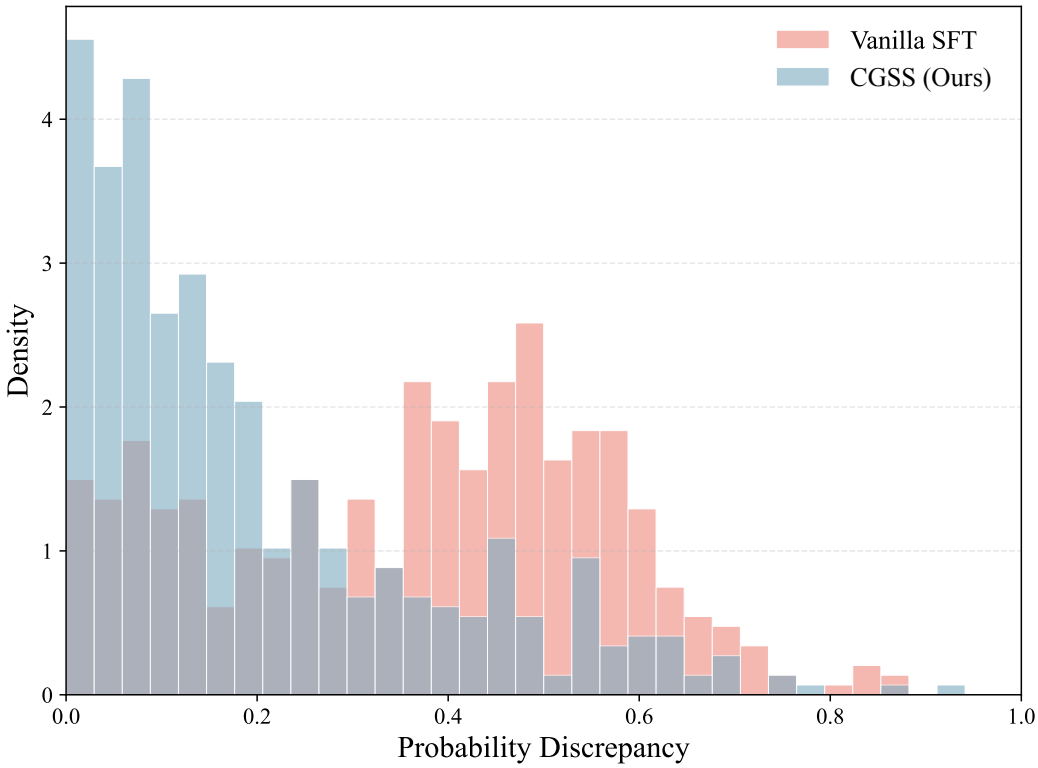


Figure 5: **Distribution of Step-Level Confidence Discrepancy.** We plot absolute differences between teacher and student confidence scores on validation CoT steps. Vanilla SFT (red) has more mass at larger gaps, while CGSS (blue) shows smaller discrepancies.

E Evaluation Benchmarks

We evaluate mathematical and scientific reasoning using five representative benchmarks. ***AIME24&25*** (AIME, 2025) contains high-difficulty competition problems from the American Invitational Mathematics Examination 2024 and 2025. ***MATH500*** (Hendrycks et al., 2021) contains 500 problems sampled from the MATH test set across seven mathematical domains. ***Olympiad-Bench*** (He et al., 2024) consists of Olympiad-level mathematical problems. ***GPQA-Diamond*** (Rein et al., 2023) is a graduate-level benchmark of expert-written multiple-choice questions spanning physics, chemistry, and biology.

F Training Details

We implement CGSS using the Llama-factory training framework. All models are trained using the configurations detailed in Table 3. We utilize the AdamW optimizer with a cosine learning rate

scheduler. To ensure fair comparisons and stable training, we adopt a unified set of hyperparameters across all backbone models. Specifically, all models are trained for 6 epochs with a global batch size of 32 and a maximum sequence length (cutoff length) of 32,768 tokens. We utilize a consistent learning rate of 5e-5 and a warmup ratio of 0.1 across all experiments.

Table 3: **Training configuration for different backbone models.**

Hyperparameter	Qwen3-4B-Base	Qwen3-8B-Base	Qwen3-4B-Instruct-2507	Qwen2.5-7B-Instruct
Learning Rate	5e-5	5e-5	5e-5	5e-5
Cutoff Length	32k	32k	32k	32k
Epochs	6	6	6	6
Batch Size	32	32	32	32
LR Scheduler	cosine_with_min_lr	cosine_with_min_lr	cosine_with_min_lr	cosine_with_min_lr
Min LR	1e-5	1e-5	1e-5	1e-5
Warmup Ratio	0.1	0.1	0.1	0.1

G Additional Experimental Results

In addition to the main baselines, we evaluate two additional selection strategies to further validate the robustness of our criteria: (1) **ESSP** (Entropy-Suppressed Supervision Prioritization): The reverse of EDSP. This method performs low-entropy selection by emphasizing steps with lower predictive entropy during training, effectively prioritizing high-confidence (often simpler) steps. (2) **HTST** (Hard-Target Supervision Tailoring): The reverse of TDST. This method performs hard-fit selection by prioritizing steps with lower length-normalized likelihood (equivalently, higher perplexity) under the student model, tailoring supervision toward harder-to-fit steps. Table 4 presents the detailed performance comparison, including these additional baselines, across all evaluated models.

G.1 Evaluation with an Alternative Trace Generator

Our sentence-level segmentation uses standard punctuation and is not tailored to DeepSeek-generated responses. To further examine whether the proposed selection criterion generalizes beyond a single trace generator, we conduct an additional experiment using gpt-oss-120B as the trace generator under the same training and evaluation settings. As shown in Table 5, CGSS consistently improves over Vanilla SFT across all evaluated benchmarks, suggesting that the benefit of confidence-gated step selection is not specific to the original trace generator.

H Limitations

Our study focuses on long-CoT mathematical and reasoning benchmarks with commonly used open-weight backbone models, which provides a controlled setting for analyzing directional divergence and step-level filtering. While the evaluated benchmarks cover diverse reasoning formats, future work may further examine broader domains, additional task families, and larger model scales to better understand how the same principle transfers across different data distributions. The theoretical result concerns a sampled-token logit coordinate, while the step-level criterion is a practical proxy evaluated through accuracy and confidence discrepancies. These measurements do not directly establish agreement between full parameter-gradient vectors.

Table 4: **Comprehensive performance comparison across multiple models and baselines.** We report the accuracy (%) on reasoning benchmarks. **Bold** indicates the best result.

Model	Method	AIME24	AIME25	MATH500	OlympiadBench	GPQA	Avg.
Qwen2.5 7B-Instruct	+ Vanilla SFT	24.20	21.60	79.80	44.20	33.10	40.60
	+ Random	15.00	20.00	80.90	43.30	31.60	38.20
	+ ESSP	18.30	16.70	**84.50**	44.40	34.10	39.60
	+ EDSP	15.00	16.70	80.30	44.10	31.30	37.50
	+ HTST	23.30	23.30	80.60	43.70	33.30	40.80
	+ TDST	16.70	23.30	81.90	43.00	34.10	39.80
	+ CGSS (Ours)	**27.50**$_{\uparrow 3.3}$	**25.00**$_{\uparrow 3.4}$	83.10$_{\uparrow 3.3}$	**45.20**$_{\uparrow 1.0}$	**34.70**$_{\uparrow 1.6}$	**43.10**$_{\uparrow 2.5}$
Qwen3 4B-Base	+ Vanilla SFT	35.00	26.70	87.00	50.70	43.80	48.60
	+ Random	33.30	25.00	86.20	49.20	41.90	47.10
	+ ESSP	29.20	30.80	86.90	51.50	46.30	48.90
	+ EDSP	31.70	28.30	85.90	51.30	45.50	48.50
	+ HTST	33.30	**31.70**	86.10	50.20	43.80	49.00
	+ TDST	37.50	26.70	86.00	50.60	47.50	49.70
	+ CGSS (Ours)	**42.50**$_{\uparrow 7.5}$	31.70$_{\uparrow 5.0}$	**88.20**$_{\uparrow 1.2}$	**52.70**$_{\uparrow 2.0}$	**47.90**$_{\uparrow 4.1}$	**52.60**$_{\uparrow 4.0}$
Qwen3 8B-Base	+ Vanilla SFT	45.00	29.20	88.20	54.70	52.90	54.00
	+ Random	41.70	26.70	88.80	51.70	51.40	52.10
	+ ESSP	47.50	29.20	89.00	55.30	55.40	55.30
	+ EDSP	44.20	31.70	89.40	54.80	54.90	55.00
	+ HTST	41.70	30.80	88.50	55.30	53.70	54.00
	+ TDST	45.00	32.50	89.00	54.00	53.80	54.90
	+ CGSS (Ours)	**50.00**$_{\uparrow 5.0}$	**36.70**$_{\uparrow 7.5}$	**89.60**$_{\uparrow 1.4}$	**55.70**$_{\uparrow 1.0}$	**56.90**$_{\uparrow 4.0}$	**57.80**$_{\uparrow 3.8}$
Qwen3 4B-Instruct	+ Vanilla SFT	51.70	37.50	83.90	54.40	51.10	55.70
	+ Random	50.00	35.00	81.30	54.80	47.90	53.80
	+ ESSP	58.30	42.50	75.40	58.40	46.20	56.20
	+ EDSP	52.50	43.30	88.60	57.60	53.20	59.00
	+ HTST	56.70	41.70	88.50	56.70	50.80	58.90
	+ TDST	49.20	43.30	85.40	57.10	53.70	57.70
	+ CGSS (Ours)	**58.30**$_{\uparrow 6.6}$	**45.00**$_{\uparrow 7.5}$	**91.40**$_{\uparrow 7.5}$	**59.80**$_{\uparrow 5.4}$	**55.40**$_{\uparrow 4.3}$	**62.00**$_{\uparrow 6.3}$

Table 5: **Generalization experiment with gpt-oss-120B as the teacher.** We report accuracy (%) on reasoning benchmarks.

Method	AIME24	AIME25	MATH500	OlympiadBench	GPQA-D	Avg.
Vanilla SFT	32.50	29.20	77.80	46.20	43.10	45.80
CGSS (Ours)	**40.70**	**30.80**	**84.00**	**49.60**	**46.50**	**50.30**