

MAKING COMPLEX REASONING STUDENT-FRIENDLY: A HYBRID LLM-TO-SLM DISTILLATION FRAMEWORK

Yongjin Yang^{1,2} Yinghui He³ Jiarui Liu⁴ Zhijing Jin^{1,2,5}
 University of Toronto¹ Vector Institute² Princeton University³
 Carnegie Mellon University⁴ MPI for Intelligent Systems⁵
 {yjyang, zjin}@cs.toronto.edu

ABSTRACT

Distilling the reasoning capabilities of large language models into smaller ones remains challenging. Off-policy distillation imitates fixed teacher trajectories and often suffers from teacher–student misalignment, leading to superficial pattern memorization. In contrast, on-policy distillation improves alignment by relying on student-generated solutions with teacher feedback, but struggles with exploration and frequently fails on complex problems when the student cannot generate valid solutions from scratch. To bridge this gap, we propose Student-Friendly Distillation (**SFD**), a framework that synergizes off-policy teacher trajectories with on-policy student generations. Specifically, **SFD** performs hybrid teacher-student generation where, starting from a teacher prefix, it switches to on-policy student generation once the student is sufficiently confident and aligned. The timing of this transition is governed by a dynamic hand-off criterion based on the student’s token entropy and its negative log-likelihood of the teacher’s reasoning tokens. After completion, the student rewrites the full solution to retain the teacher’s high-level reasoning in its own style, yielding trajectories that preserve quality while remaining in-distribution. Extensive experiments across six reasoning benchmarks demonstrate that **SFD** consistently outperforms both off-policy distillation and the on-policy rejection fine-tuning method, without access to teacher log probabilities, which are costly to compute.

1 INTRODUCTION

The scaling of data and model size has significantly increased large language models (LLMs)’ reasoning capabilities (DeepSeek-AI et al., 2025; OpenAI, 2024; Comanici et al., 2025). However, utilizing these massive models in real-world applications remains challenging due to their prohibitive inference costs and high latency. Consequently, enabling small language models (SLMs) to perform robust reasoning within a constrained computational budget has garnered significant research attention.

Distillation remains the predominant approach for compressing model capabilities (Hinton et al., 2015; Kim and Rush, 2016; Sanh et al., 2019; Ho et al., 2023; Ko et al., 2024). Off-policy approaches minimize the divergence between student and teacher distributions on fixed teacher-generated sequences (Kim and Rush, 2016; Ho et al., 2023; Hsieh et al., 2023). In contrast, on-policy approaches involve the teacher providing feedback on student-generated trajectories or sometimes jointly generating responses to be trained (Agarwal et al., 2024; Xu et al., 2025b).

However, naively distilling reasoning traces into smaller LLMs often yields suboptimal results (Xu et al., 2025a). The primary bottleneck is teacher-student misalignment: the teacher’s outputs frequently lie outside the student’s distribution, causing the student to memorize complex responses rather than generalizing the underlying logic. Although on-policy approaches have been proposed to address this (Kim et al., 2025; Agarwal et al., 2024; Xu et al., 2025b), such fine-grained, token-level collaboration with the teacher can disrupt the coherence of high-level reasoning and often faces exploration challenges. Furthermore, relying on access to the teacher’s log-probabilities is often computationally infeasible for large models generating long sequences and, critically, impossible for proprietary models. These challenges motivate the development of a flexible distillation framework

that ensures teacher-student alignment without relying on dense teacher signals, while leveraging the high-level reasoning of the teacher.

Contribution. In this paper, we propose Student-Friendly Distillation (**SFD**), a distillation framework that bridges the capability gap between LLMs and SLMs without sacrificing reasoning coherence. The core idea is to leverage a dynamic hybrid generation strategy: utilizing a teacher-provided off-policy prefix to guide high-level planning, then transitioning to student generation once the student exhibits sufficient confidence. This is followed by a rewrite step to fully align the solution with the student’s distribution. Our specific contributions are as follows:

1. **Analysis of Student Alignment and Prefix Effects.** We conduct a token-level analysis of teacher-student alignment on reasoning datasets (§2.2). We observe that the student’s high uncertainty and misalignment are disproportionately concentrated in the initial reasoning steps (the prefix), where high-level planning typically occurs. We also demonstrate that providing the student with even a short teacher prefix significantly improves its ability to generate correct solutions from scratch.
2. **Prefix-based Hybrid Generation and Student Rewrite.** Building on these insights, we propose **SFD**, a method that synergizes off-policy teacher data with on-policy student generation. We introduce a dynamic hand-off mechanism that monitors the student’s smoothed entropy and negative log-likelihood (NLL) on the teacher’s response (§3.1). This allows the student to take over generation exactly when it becomes confident, leveraging the teacher’s high-level logic while ensuring the completion is within its own capabilities. Additionally, we employ a student rewrite step (§3.2) to project the teacher’s reasoning path into the student’s native writing style, effectively decoupling reasoning correctness from stylistic mimicry.
3. **Superior Performance and Efficiency.** Extensive experiments demonstrate that **SFD** consistently outperforms standard off-policy distillation with supervised fine-tuning (SFT) and rejection fine-tuning (RFT) baselines across diverse math and STEM benchmarks (§4.2). Crucially, **SFD** achieves these results without requiring access to the teacher’s internal states (*e.g.*, log-probabilities) or complex reinforcement learning RL optimization. The method remains highly efficient, relying on standard supervised fine-tuning that converges rapidly.

2 BACKGROUND

In this section, we first establish the formal foundations of knowledge distillation and fine-tuning in reasoning tasks (§2.1). Subsequently, we motivate our proposed approach, Student-Friendly Distillation (**SFD**), by analyzing the dynamics of student uncertainty and alignment during the generation process, demonstrating how the teacher’s initial reasoning steps (the “prefix”) serve as critical scaffolding for the student (§2.2).

2.1 PRELIMINARIES

We operate within the context of reasoning-intensive tasks where a query q requires a multi-step rationale z to reach a final answer y . We denote the full response as $r = (z, y)$. Let π_θ denote the fixed teacher policy (LLM) and π_ϕ denote the student policy (SLM) we wish to optimize.

Supervised Fine-Tuning (SFT). Knowledge distillation in LLMs is predominantly formulated as Supervised Fine-Tuning (SFT) on teacher-generated traces. The objective is to minimize the forward KL divergence $D_{\text{KL}}(\pi_\theta \parallel \pi_\phi)$, which is equivalent to maximizing the log-likelihood of the student generating the teacher’s rationale z_{teacher} :

$$\mathcal{L}_{\text{SFT}}(\phi) = -\mathbb{E}_{q \sim \mathcal{D}, r \sim \pi_\theta(\cdot | q)} [\log \pi_\phi(r | q)]. \quad (1)$$

While effective for simple tasks, this objective forces the student π_ϕ to cover the entire support of the teacher π_θ . When the teacher’s reasoning distribution lies outside the student’s effective support (a phenomenon often termed “distributional shift”), this leads to *mode-covering* behavior where the student mimics surface-level tokens without grasping the underlying semantic logic.

Rejection Fine-Tuning (RFT). To mitigate distribution shift, Rejection Fine-Tuning (RFT) leverages the student’s own distribution (Zhang et al., 2023; Zelikman et al., 2022). For a query q , k responses are sampled from the student: $\{r_1, \dots, r_k\} \sim \pi_\phi(\cdot | q)$. A verification oracle (*e.g.*, ground

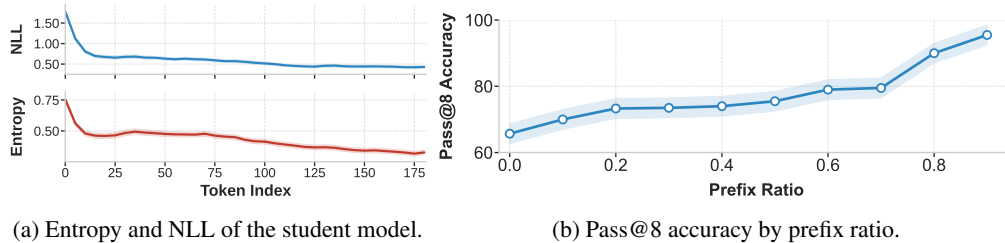


Figure 1: **(a) NLL and Uncertainty Dynamics:** Token-level entropy and NLL of student ($Q_{wen2.5-3B}$) on 1k teacher traces ($Q_{wen3-32B}$). Downward trends indicate the student converges toward teacher alignment (lower NLL) and higher confidence (lower entropy) as the sequence progresses. **(b) Prefix Sensitivity:** Student accuracy vs. prefix ratio on MATH-SFT. A 20% prefix yields substantial gains, followed by a plateau and a secondary performance boost at 80%.

truth) filters for correctness, yielding a set $\mathcal{R}_{\text{correct}}$. The student is updated only on its own successful trajectories:

$$\mathcal{L}_{\text{RFT}}(\phi) = -\mathbb{E}_{r \in \mathcal{R}_{\text{correct}}} [\log \pi_{\phi}(r | q)]. \quad (2)$$

While RFT ensures distribution alignment, it suffers from sample inefficiency and limited exploration, often preventing the student from discovering complex reasoning paths needed for hard problems and capping performance.

2.2 MOTIVATING OBSERVATION

We present empirical evidence suggesting that the primary bottleneck in student reasoning lies in the initial planning phase, thereby motivating the prefix-based intervention incorporated into **SFD** (§3).

Entropy and Alignment Analysis. We analyze the token-level dynamics of a $Q_{wen2.5-3B}$ student model evaluating solutions generated by a $Q_{wen3-32B}$ teacher. Figure 1a illustrates the student’s entropy and negative log-likelihood (NLL) over the teacher’s tokens, smoothed via a sliding window of 15 tokens with a stride of 5.

We observe a distinct downward trend in both NLL and entropy as the token index increases. This indicates that the student’s uncertainty and misalignment are concentrated in the early prefix—typically the phase requiring high-level planning and problem formulation. However, once this initial context is established, the student’s confidence grows, and its alignment with the teacher improves significantly. This suggests that while the student struggles to *initiate* complex reasoning strategies, it is capable of effectively *executing* them when guided by a brief teacher prefix.

The Critical Role of the Prefix. To quantify the value of this prefix guidance, we evaluate the student’s performance when conditioned on varying ratios of the teacher’s response before the student completes the answer. As illustrated in Figure 1b, even a short teacher prefix (20%) yields an increase in pass@8 accuracy—a metric chosen to verify whether the added context enables the student to overcome exploration bottlenecks and recover correct reasoning paths. This suggests that a significant portion of the teacher-student “reasoning gap” stems from the initial high-level planning stage: minimal structural guidance enables the student to correctly execute the subsequent steps independently. Crucially, this low-prefix regime preserves outputs that are largely student-aligned while still benefiting from improved reasoning. In contrast, providing a heavily extended prefix (*e.g.*, $\sim 80\%$) yields further accuracy gains but forces the completion to converge toward the teacher’s distribution, thereby diminishing the student’s stylistic and distributional autonomy.

3 METHOD

In this paper, we propose Student-Friendly Distillation (**SFD**), a framework designed to bridge the capability gap between a strong teacher and a weaker student without incurring the drawbacks of pure off-policy or on-policy distillation. **SFD** consists of two phases: (1) **Hybrid Teacher-Student Generation** (§3.1), which dynamically determines the optimal point to transfer control from teacher to student, and (2) **Student Rewrite** (§3.2), which projects the resulting solution onto the student’s

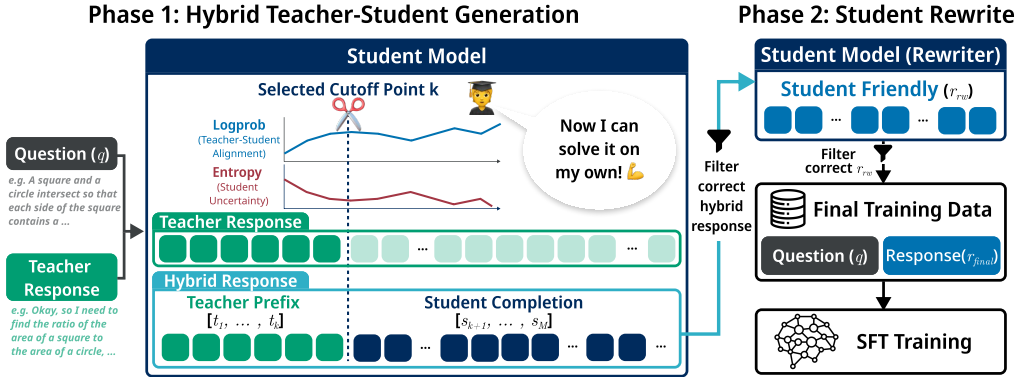


Figure 2: Overview of **SFD**. The framework operates in two phases to effectively transfer reasoning capabilities from a large teacher to a smaller student. **(Left) Hybrid Teacher-Student Generation:** The process begins with a teacher prefix that guides the generation. We dynamically monitor the student’s alignment (NLL) and confidence (Entropy) using a sliding window. Once the student demonstrates sufficient competence (low NLL or low Entropy), control is handed off to the student to complete the reasoning path, resulting in a hybrid trajectory r_{hybrid} . **(Right) Student Rewrite:** To mitigate distributional shift, the student rewrites the hybrid trajectory into its own natural language distribution (r_{rw}). This projects the correct reasoning logic onto the student’s native manifold before supervised fine-tuning.

native manifold. An overview of our method is illustrated in Figure 2, and the full algorithm is presented in Algorithm 1 in Appendix C.

3.1 HYBRID TEACHER-STUDENT GENERATION

While the teacher’s prefix provides a critical signal for high-level reasoning, determining the optimal prefix length involves a delicate trade-off: a prefix that is too short may fail to guide the student, whereas one that is too long exacerbates teacher-student misalignment. To address this, we introduce a dynamic hand-off strategy governed by two distinct signals: *Misalignment* (how surprising the teacher’s token is to the student) and *Uncertainty* (how unsure the student is of its own prediction).

Let $r^* = (t_1, \dots, t_T)$ be the teacher’s reference solution (off-policy data) for a query q . At each step i , we compute:

1. **Misalignment (\mathcal{L}_i):** The negative log-likelihood (NLL) of the teacher’s token under the student model: $\mathcal{L}_i = -\log \pi_\phi(t_i | q, t_{<i})$.
2. **Uncertainty (\mathcal{H}_i):** The entropy of the student’s next-token distribution: $\mathcal{H}_i = \mathbb{H}[\pi_\phi(\cdot | q, t_{<i})]$.

To ensure robustness against token-level noise, we apply a sliding window average $S(\cdot)$ of size W over these metrics. The student is considered competent to take over generation at step i if it exhibits either high alignment (low NLL) or high confidence (low entropy):

$$\text{Ready}(i) = \mathbb{I}[S(\mathcal{L}_i) < \tau_{\text{align}}] \vee \mathbb{I}[S(\mathcal{H}_i) < \tau_{\text{conf}}]. \tag{3}$$

Crucially, the condition $S(\mathcal{H}_i) < \tau_{\text{conf}}$ permits the student to assume control even if it diverges from the teacher, provided the student is sufficiently confident in its own reasoning path.

We define the optimal **cut-off point** k as the start of the first *sustained* sequence of valid states of length P :

$$k = \min\{t \mid \forall j \in [t, t + P], \text{Ready}(j) = 1\}. \tag{4}$$

This prevents handing off to the student due to a transient spike in log probability or low entropy. Using this adaptive k , we generate a **hybrid response** r_{hybrid} :

$$r_{\text{hybrid}} = r_{[1:k]}^* \oplus \text{Sample}(\pi_\phi(\cdot | x, r_{[1:k]}^*)). \tag{5}$$

Note that we sample N responses per prefix. This hybrid generation ensures that the teacher guides the student through high-uncertainty regions (the “hard parts”) and releases control once the student enters its zone of competence. We then select the one correct response to finalize the r_{hybrid} .

3.2 STUDENT REWRITE

While r_{hybrid} represents a valid reasoning path, it is a composite sequence. The prefix $r_{[1:k]}^*$ originates from the teacher’s distribution, which may contain stylistic or vocabulary choices that are out-of-distribution for the student. Training directly on r_{hybrid} can, therefore, destabilize the student’s language model.

To address this, we treat r_{hybrid} not as a ground-truth target, but as a semantic scaffold. We employ a rewrite step to translate this hybrid solution into the student’s native distribution. Specifically, we sample N rewritten solutions from the student conditioned on the scaffold:

$$r_{\text{rw}} \sim \pi_{\phi}(\cdot \mid q, \text{context} = r_{\text{hybrid}}), \quad (6)$$

where the resulting r_{rws} are then each verified against the ground-truth oracle, retaining only correct trajectories, denoted as r_{final} . These r_{final} are accumulated to form the final supervised fine-tuning dataset by pairing them with the query q . This process projects the teacher’s correct reasoning onto the student’s native generation manifold, preserving logical correctness while avoiding stylistic mimicry. Prompts for rewriting are provided in Appendix B, and full hyperparameters for **SFD** are detailed in Appendix D.1.

3.3 INTERPRETATION OF **SFD**

SFD decouples reasoning exploration from distributional alignment. Ideally, we maximize the student π_{ϕ} ’s likelihood over the set of semantically correct chains \mathcal{R}^* : $\max_{\phi} \log \int_{r \in \mathcal{R}^*} \pi_{\phi}(r \mid q) dr$. Standard SFT approximates this with a single $r^* \in \mathcal{R}^*$, causing high-variance gradients if $\pi_{\phi}(r^*) \approx 0$ (the “out-of-support” problem).

Phase 1: Entropy Reduction via Hybrid Generation. We decompose entropy at split point k :

$$H_{\pi_{\phi}}(r) = \underbrace{H_{\pi_{\phi}}(r_{<k})}_{\text{Prefix Entropy}} + \underbrace{H_{\pi_{\phi}}(r_{\geq k} \mid r_{<k})}_{\text{Suffix Entropy}} \quad (7)$$

Failures primarily stem from *Prefix Entropy* (planning uncertainty). Substituting the teacher’s prefix $r_{<k}^*$ collapses this term to zero. Sampling from the resulting low-entropy distribution $\pi_{\phi}(\cdot \mid q, r_{<k}^*)$ allows exploring \mathcal{R}^* without the initial planning bottleneck.

Phase 2: Manifold Projection via Rewrite. Valid hybrid responses ($r_{\text{hybrid}} \in \mathcal{R}^*$) may retain divergent teacher artifacts. Defining the student’s effective manifold as $\mathcal{M}_{\phi} = \{r : \pi_{\phi}(r \mid q) > \epsilon\}$, the rewrite step \mathcal{P}_{ϕ} projects the solution into the student’s distribution:

$$r_{\text{final}} \approx \arg \min_{r \in \mathcal{R}^*} D_{\text{KL}}(\delta_r \parallel \pi_{\phi}(\cdot \mid q)) \quad (8)$$

Training on r_{final} ensures logical correctness ($r \in \mathcal{R}^*$) within native support (\mathcal{M}_{ϕ}), minimizing the distributional shift found in SFT.

4 EXPERIMENTS

We evaluate the efficacy of **SFD** guided by the following research questions:

- RQ1: Performance Efficacy:** How does **SFD** compare with standard off-policy SFT and on-policy self-distillation (RFT) on reasoning benchmarks?
- RQ2: Component Analysis:** How do the *Hybrid Generation* and *Student Rewrite* phases individually contribute to reasoning accuracy and student alignment?
- RQ3: Mechanism Validity:** Is the proposed dynamic hand-off mechanism—based on student confidence and alignment—more effective than static or heuristic-based prefix selection strategies?

4.1 EXPERIMENTAL SETTINGS

Datasets. We utilize the MATH-SFT dataset from Huan et al. (2025), which aggregates queries from the MATH benchmark (Hendrycks et al., 2021) and DeepScaler (Luo et al., 2025). The dataset

Table 1: Average accuracy (%) and standard deviation on mathematical reasoning benchmarks using Qwen2.5-3B, Qwen2.5-1.5B, and Llama3.2-3B. We compare Zero-shot CoT, RFT, SFT, and our proposed method **SFD** across MATH500, AIME (2024, 2025), AMC23, GPQA-Diamond, and Minerva. Rows corresponding to **SFD** are highlighted in blue, and the best scores are bolded.

Model	Method	MATH500	AIME2024	AIME2025	AMC23	GPQA-Diamond	Minerva	Average
Qwen2.5-3B	CoT	59.00 \pm 1.06	4.58 \pm 5.62	0.42 \pm 1.18	27.50 \pm 2.89	25.88 \pm 2.35	14.89 \pm 2.17	22.04 \pm 0.45
	RFT	61.95 \pm 0.85	5.00 \pm 2.52	1.00 \pm 0.00	34.38 \pm 2.39	28.03 \pm 2.20	16.27 \pm 1.05	24.44 \pm 0.68
	SFT	53.75 \pm 2.41	3.75 \pm 2.14	2.08 \pm 2.48	31.88 \pm 4.27	26.27 \pm 2.67	15.26 \pm 1.71	22.16 \pm 1.17
	SFD	62.10 \pm 1.47	7.92 \pm 3.05	3.75 \pm 2.14	35.62 \pm 8.75	30.56 \pm 2.90	16.45 \pm 0.70	26.07 \pm 1.80
Qwen2.5-1.5B	CoT	38.25 \pm 2.32	0.83 \pm 1.54	0.83 \pm 1.54	15.62 \pm 1.25	25.25 \pm 2.06	6.89 \pm 2.00	14.61 \pm 0.60
	RFT	46.35 \pm 0.93	2.50 \pm 3.88	0.42 \pm 1.18	26.88 \pm 5.54	22.85 \pm 1.12	5.52 \pm 1.27	17.42 \pm 1.65
	SFT	40.65 \pm 1.38	0.83 \pm 1.54	0.83 \pm 1.54	18.12 \pm 3.15	26.64 \pm 1.62	9.10 \pm 1.32	16.03 \pm 0.55
	SFD	47.40 \pm 0.43	2.92 \pm 3.30	1.67 \pm 2.52	21.25 \pm 4.33	26.26 \pm 1.65	9.74 \pm 1.26	18.21 \pm 1.29
Llama3.2-3B	CoT	34.95 \pm 0.34	2.50 \pm 2.96	0.00 \pm 0.00	20.00 \pm 3.54	27.78 \pm 2.37	8.99 \pm 0.88	15.70 \pm 0.50
	RFT	42.80 \pm 1.10	4.17 \pm 2.96	0.42 \pm 1.18	19.38 \pm 2.39	21.90 \pm 0.21	13.14 \pm 2.41	16.97 \pm 0.87
	SFT	42.00 \pm 2.01	2.08 \pm 2.48	0.42 \pm 1.18	19.38 \pm 3.15	29.29 \pm 2.30	9.65 \pm 1.18	17.14 \pm 1.01
	SFD	46.25 \pm 1.68	5.83 \pm 4.96	0.83 \pm 1.54	28.12 \pm 4.27	27.40 \pm 2.31	11.03 \pm 2.38	19.91 \pm 1.14

provides teacher trajectories generated by Qwen3-32B (Yang et al., 2025), containing both reasoning traces and final answers. For computational efficiency, we sample the first 1,000 instances from the 40k dataset for training.

For evaluation, we employ a diverse suite of reasoning benchmarks: MATH500 (Hendrycks et al., 2021), AIME 2024 & 2025 (Art of Problem Solving, 2025), AMC 23 (MAA, 2023), and Minerva Lewkowycz et al. (2022) for mathematical reasoning, as well as GPQA-Diamond (Rein et al., 2024) for broader STEM capabilities. To ensure statistical robustness, we report the average accuracy over multiple independent inference runs: 8 runs for AIME and 4 runs for the others.

Models. We employ Qwen3-32B (Yang et al., 2025) as the fixed teacher model. For student models, we primarily use Qwen2.5-3B (Yang et al., 2024a), since the newer Qwen3 series has undergone extensive mid-training on similar mathematical data, which could confound distillation gains. We additionally include Qwen2.5-1.5B to study the effect of model scale, and we evaluate Llama3.2-3B (Dubey et al., 2024) to assess cross-family generalizability. All models are instruction-tuned variants, but we omit this designation in the model names for brevity.

Implementation. Training is performed using the `trl` library. We fine-tune the models for 5 epochs with a learning rate of $5e-6$, a weight decay of $1e-4$, and a global batch size of 32, utilizing 4 NVIDIA H100 GPUs. Training takes less than an hour. Inference is conducted via vLLM (Kwon et al., 2023) with a temperature of 1.0 and top- p of 0.95 to facilitate exploration during evaluation with multiple repetitions. Detailed prompts and hyperparameter configurations are provided in Appendices B and D.2.

Baselines. For the baselines, we compare **SFD** with zero-shot Chain-of-Thought prompting (Wei et al., 2022) using the original models (*CoT*), naive supervised fine-tuning with teacher-generated responses (*SFT*), rejection fine-tuning with 8 responses sampled per query (*RFT*). We exclude logit-based distillation methods (e.g., GKD (Agarwal et al., 2024), SKD (Xu et al., 2025b)) as they are computationally infeasible with the Qwen3-32B teacher due to the prohibitive cost of computing teacher log-probabilities for every student token.

4.2 MAIN RESULTS

Table 1 compares **SFD** with baselines across three student architectures. **SFD** consistently outperforms the strongest baseline (RFT or SFT) on all benchmarks, improving average accuracy by approximately 2% on Qwen2.5-3B and Llama3.2-3B, and 1% on Qwen2.5-1.5B. These gains are notable given that the models are already highly instruction-tuned.

Standard SFT yields marginal gains for the Qwen series due to student-teacher distributional divergence, though it aids Llama given its lower base performance. While RFT improves over zero-shot CoT via on-policy training, it lacks external exploration. In contrast, **SFD** delivers a consistent $\sim 2\%$ improvement by leveraging the teacher’s high-level reasoning for exploration while ensuring final solutions remain aligned with the student’s distribution.

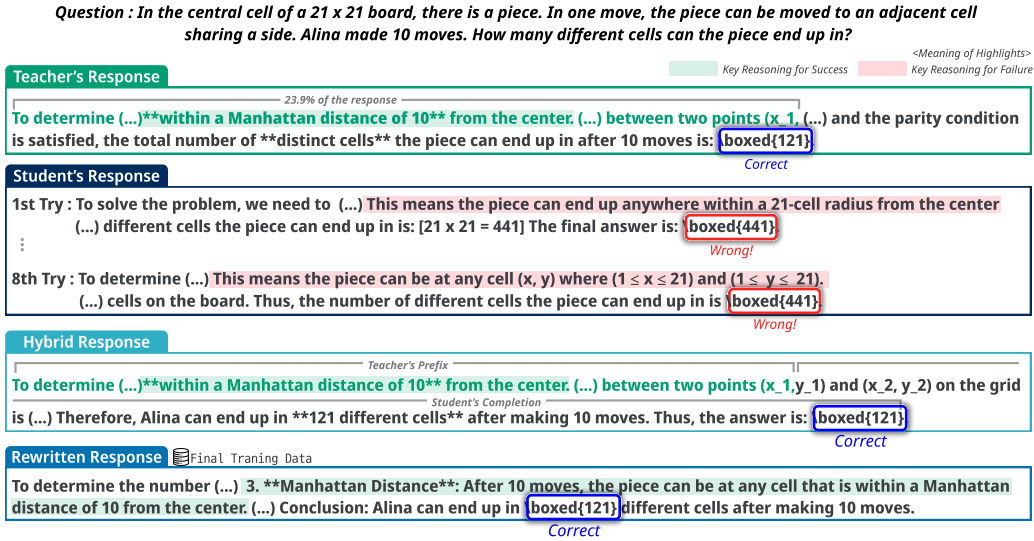


Figure 3: Qualitative example of **SFD**-generated training data: the base student fails all 8 attempts, but a short teacher prefix (23.9%, selected via entropy-based truncation) primes the correct distance metric and enables success. The student rewrite step then removes teacher artifacts and rewrites the trajectory in the student’s style while preserving correctness.

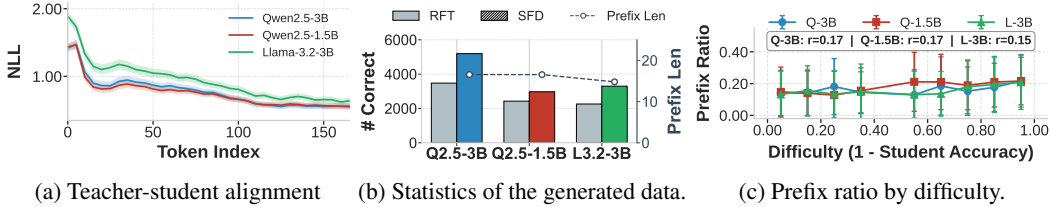


Figure 4: **(a)** Smoothed NLL loss by token index for Qwen2.5-3B, Qwen2.5-1.5B, and Llama3-3B on teacher traces (as in Figure 1a). Llama3-3B exhibits the poorest alignment because the teacher is from the Qwen model series. **(b)** Statistics of data generated using **SFD**. “# Correct” denotes the number of correct responses over 1000 samples, compared with self-generation during RFT. The right y-axis shows the average prefix length used by **SFD** for each model, which centers around 16%. **(c)** Prefix ratio relative to question difficulty. Difficulty is defined as the accuracy of each question when sampled 8 times (r denotes correlation). The selected prefix length shows a slight correlation with the student’s question difficulty.

SFD shows significant gains on challenging AIME benchmarks, surpassing RFT by 2.92% on Qwen2.5-3B. This suggests **SFD** provides high-quality solutions within the student’s capability range. Gains on the 1.5B model are smaller (0.84%) likely due to weaker rewriting abilities. Improvements also extend to broader STEM performance (GPQA-Diamond).

Finally, **SFD** is highly effective for Llama3.2-3B despite significant teacher–student misalignment (Figure 4a), achieving over 10% absolute improvement against zero-shot CoT. This confirms **SFD** successfully transforms high-level teacher reasoning into student-friendly distillation data. Qualitative examples are provided in Appendix E.2.

4.3 ANALYSIS

Data Statistics. Figure 4b summarizes statistics for **SFD** (8 samples per 1k queries). Compared to RFT, **SFD** increases the number of correct responses by 0.5k–2k, demonstrating that the teacher prefix effectively boosts reasoning. The average prefix ratios remain low: 16.56% (Qwen2.5-3B), 16.16% (Qwen2.5-1.5B), and 14.56% (Llama3.2-3B). This confirms that **SFD** leverages high-level teacher guidance while largely preserving the student’s intrinsic behavior.

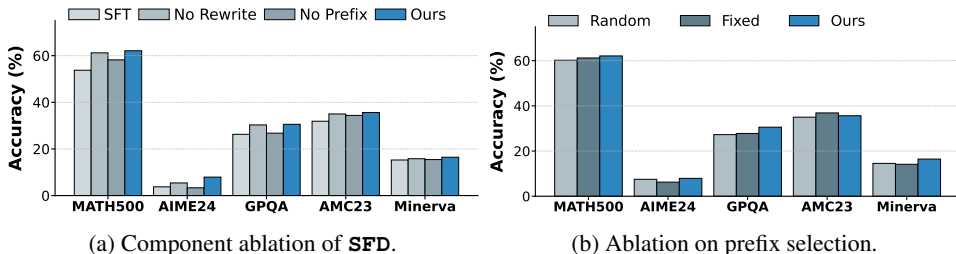


Figure 5: **Ablation studies on MATH500, AIME2024, GPQA, AMC23, and Minerva.** (a) **Component Analysis:** Both *hybrid teacher-student generation* and *student rewrite* are essential; removing either leads to a performance drop across all benchmarks. (b) **Prefix Selection:** Comparison of our uncertainty/NLL-based method against fixed (20%) and random (10%–50%) lengths. While **SFD** performs best, fixed/random baselines are competitive due to comparable average prefix lengths (20%–25% vs. **SFD**’s 16.56%).

Figure 4c analyzes prefix usage by difficulty. We observe a weak positive correlation where students use slightly more prefix for harder questions. However, ratios remain low even for difficult queries. This behavior is desirable, as it ensures the student benefits from guidance without over-relying on the teacher or drifting off-manifold.

Sample Analysis. Figure 3 illustrates a distillation example from **SFD** where the base student fails in all eight attempts due to an incorrect geometric assumption, treating the reachable region as an axis-aligned square rather than respecting the diamond-shaped Manhattan-distance constraint. Injecting the first 23.9% of the teacher’s response—selected via entropy analysis—primes the student with the correct metric and framing, enabling it to recover the correct reasoning trajectory and reach the correct answer. Although the teacher prefix introduces stylistic artifacts outside the student’s distribution, the *Student Rewrite* step maps the mixed trajectory back to the student’s style, explicitly incorporating the Manhattan-distance constraint and producing higher-quality, well-aligned training data. More sample analyses with different models are available in Appendix E.

4.4 ABLATION STUDIES

We perform ablations using Qwen2.5–3B to isolate **SFD** design choices.

Ablation on Each Component. Figure 5a compares *Hybrid Generation (No Rewrite)* and *Student Rewrite (No Prefix)*. While both variants outperform SFT, removing either lowers performance. Notably, removing the prefix causes a larger drop (generally $> 1\%$), as rewriting alone cannot fully match the student distribution. *No Rewrite* fares slightly better since generation remains student-driven. Full **SFD** yields the best performance, confirming the synergy of both components.

Ablation on Prefix Selection. Figure 5b compares our dynamic strategy (entropy/NLL) against *fixed* (ratio 0.2) and *random* (uniform 0.1–0.5) selection. The dynamic approach consistently outperforms these heuristics, which fail to adapt to query complexity or introduce misalignment. By detecting student confidence, **SFD** maximizes teacher utility while minimizing out-of-distribution exposure. While *fixed* and *random* remain competitive, the dynamic mechanism eliminates manual tuning, offering superior generalizability across models and tasks.

5 CONCLUSION

In this work, we introduce Student-Friendly Distillation (**SFD**), a distillation framework that synergizes off-policy guidance with on-policy exploration. Specifically, **SFD** employs a dynamic hand-off mechanism that uses teacher prefixes to scaffold high-level reasoning before transitioning to student generation, followed by a rewrite step to ensure robust alignment. This approach produces distillation data that remains within the student’s manifold while leveraging higher-level reasoning. Extensive experiments demonstrate that **SFD** consistently outperforms standard off-policy distillation and rejection fine-tuning across diverse benchmarks without requiring prohibitive access to teacher log-probabilities, establishing an efficient new standard for reasoning distillation.

ACKNOWLEDGEMENT

We thank Jimin Lee for extensive discussions and for helping generate the figures.

This material is based in part upon work supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039B; by the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645; by Schmidt Sciences SAFE-AI Grant; by the Frontier Model Forum and AI Safety Fund; by Coefficient Giving; by the Canadian AI Safety Institute Research Program at CIFAR; by the Canadian AI Safety Institute Research Program at CIFAR through a Catalyst Award; by the Survival and Flourishing Fund; and by the Cooperative AI Foundation. Resources used in preparing this research project were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute.

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.
- Art of Problem Solving. AIME Problems and Solutions, 2025. URL https://artofproblemsolving.com/wiki/index.php/AIME_Problems_and_Solutions. Accessed: 2025-05-15.
- Edoardo Cetin, Tianyu Zhao, and Yujin Tang. Reinforcement learning teachers of test time scaling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL <https://openreview.net/forum?id=tebG8q5EeK>.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.
- 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, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanbiao Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen

- Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL <https://arxiv.org/abs/2501.12948>.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- Dan Hendrycks, Collin Burns, Steven Basart, and et al. Measuring mathematical problem solving with the MATH dataset. *arXiv:2103.03874*, 2021.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 14852–14882, 2023.
- Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8003–8017, 2023.
- Maggie Huan, Yuetai Li, Tuney Zheng, Xiaoyu Xu, Seungone Kim, Minxin Du, Radha Pooven-dran, Graham Neubig, and Xiang Yue. Does math reasoning improve general llm capabilities? understanding transferability of llm reasoning. *arXiv preprint arXiv:2507.00432*, 2025.
- Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo M Ponti, and Ivan Titov. Blending supervised and reinforcement fine-tuning with prefix sampling. *arXiv preprint arXiv:2507.01679*, 2025.
- Wangyi Jiang, Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. Teach small models to reason by curriculum distillation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.376. URL <https://aclanthology.org/2025.emnlp-main.376/>.
- Jaehoon Kim, Kwangwook Seo, and Dongha Lee. In their own words: Reasoning traces tailored for small models make them better reasoners. *arXiv preprint arXiv:2509.22230*, 2025.
- Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1317–1327, 2016.
- Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024.
- Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. Distillm-2: A contrastive approach boosts the distillation of llms. In *Forty-second International Conference on Machine Learning*, 2025.
- Takeshi Kojima, Shixiang Seno, Len Schubert, and et al. Large language models are zero-shot reasoners. *arXiv:2205.11916*, 2022.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626, 2023.
- Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

- Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*, 2022.
- Jiarui Liu, Kaustubh Dhole, Yingheng Wang, Haoyang Wen, Sarah Zhang, Haitao Mao, Gaotang Li, Neeraj Varshney, Jingguo Liu, and Xiaoman Pan. Toward honest language models for deductive reasoning. *arXiv preprint arXiv:2511.09222*, 2025a.
- Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. Uft: Unifying supervised and reinforcement fine-tuning. *arXiv preprint arXiv:2505.16984*, 2025b.
- Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, et al. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*, 2025. URL <https://huggingface.co/agentica-org/DeepScaleR-1.5B-Preview>.
- Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, et al. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. *arXiv preprint arXiv:2506.07527*, 2025.
- MAA. American mathematics competitions - amc. <https://maa.org/>, 2023.
- OpenAI. Openai o1 system card, September 2024. URL <https://openai.com/index/openai-o1-system-card/>. This report outlines the safety work carried out prior to releasing OpenAI o1-preview and o1-mini, including external red teaming and frontier risk evaluations according to our Preparedness Framework.
- Abhishek Panigrahi, Bingbin Liu, Sadhika Malladi, Sham Kakade, and Surbhi Goel. In good graces: Principled teacher selection for knowledge distillation. *arXiv preprint arXiv:2511.02833*, 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. In *First Conference on Language Modeling*, 2024.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Haoran Xu, Baolin Peng, Hany Awadalla, Dongdong Chen, Yen-Chun Chen, Mei Gao, Young Jin Kim, Yunsheng Li, Liliang Ren, Yelong Shen, et al. Phi-4-mini-reasoning: Exploring the limits of small reasoning language models in math. *arXiv preprint arXiv:2504.21233*, 2025a.
- Wenda Xu, Rujun Han, Zifeng Wang, Long Le, Dhruv Madeka, Lei Li, William Yang Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. In *The Thirteenth International Conference on Learning Representations*, 2025b.
- An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024a.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- Yuqing Yang, Yan Ma, and Pengfei Liu. Weak-to-strong reasoning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8350–8367, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.490. URL <https://aclanthology.org/2024.findings-emnlp.490/>.

Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*, 2025.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *Advances in Neural Information Processing Systems*, 37:333–356, 2024.

Yifan Zhang, Jingqin Yang, Yang Yuan, and Andrew Chi-Chih Yao. Cumulative reasoning with large language models. *arXiv preprint arXiv:2308.04371*, 2023.

A RELATED WORK

A.1 LLM REASONING

Chain-of-thought (CoT) reasoning allows LLMs to tackle complex tasks by generating intermediate logical steps (Wei et al., 2022; Kojima et al., 2022), a capability often enhanced through supervised fine-tuning (Ho et al., 2023; Hsieh et al., 2023) or preference learning (Zhang et al., 2024). While recent “Large Reasoning Models” (LRMs) achieve state-of-the-art performance via extensive test-time search and backtracking (often termed Long-CoT) (OpenAI, 2024; DeepSeek-AI et al., 2025; Yeo et al., 2025), these trajectories are often excessively long and complex. We therefore focus on distilling standard instruct-tuned models, as LRM paths typically lie outside the student’s effective support, making direct distillation inefficient.

A.2 KNOWLEDGE DISTILLATION

Knowledge distillation (KD) compresses capabilities by training student models to mimic a teacher (Hinton et al., 2015). **Off-policy** methods minimize divergence on fixed teacher trajectories (Kim and Rush, 2016; Sanh et al., 2019; Ho et al., 2023; Hsieh et al., 2023; Li et al., 2022). Recent work has also explored teacher selection metrics (Panigrahi et al., 2025) or curricula for transferring complex reasoning (Yang et al., 2024b; Jiang et al., 2025). However, these approaches often suffer from severe distribution mismatch, where the student memorizes the teacher’s tokens without internalizing the underlying logic (Agarwal et al., 2024).

To mitigate this, **on-policy** methods train the student on its own generations, using teacher feedback on student trajectories or on mixtures of teacher and student-generated trajectories (Agarwal et al., 2024; Ko et al., 2024; Xu et al., 2025b; Kim et al., 2025; Ko et al., 2025). Cetin et al. (2025) further proposes training the teacher for alignment. However, these methods typically require computationally expensive teacher log-probabilities and struggle with exploration when the student fails to generate valid initial paths.

Recent related RL lines of work combine RL with SFT (Ma et al., 2025; Liu et al., 2025a) and sometimes use demonstration prefixes (Liu et al., 2025b; Huang et al., 2025) to mitigate this issue. Nevertheless, these approaches typically rely on a static hand-off between off-policy and on-policy stages and require more compute-intensive, longer on-policy RL training, without addressing stylistic issues. Our framework addresses these limitations by combining teacher prefixes (to facilitate exploration) with student rewriting (to reduce misalignment), enabling robust alignment without requiring teacher log-probabilities and without constraining teacher selection.

B EXPERIMENT PROMPTS

B.1 PROMPT FOR GENERATION (INCLUDING EVALUATION)

System Prompt

You are a helpful assistant.

User Prompt

Here is the given reasoning problem. Provide the reasoning first and then the answer. Your final answer should be in `\boxed{answer}` format at the end of your response.:

{question}

B.2 STUDENT REWRITE PROMPT

System Prompt

You are a helpful assistant.

User Prompt

You are a helpful assistant.

Problem:

{question}

Here is a reference reasoning path that leads to the correct answer:

<reference>

{reference}

</reference>

Task: Solve the problem above using your own words. Use the logic in the reference to guide you, but do not “blindly copy the style and content” of the reference. Ensure the reasoning is complete and the final answer is correct.

Put your final answer within `\boxed{answer}`.

C ALGORITHM

We provide the detailed pseudocode for Student-Friendly Distillation (**SFD**) in Algorithm 1. This formulation explicitly details the sliding window smoothing for misalignment and uncertainty metrics, the fallback mechanism for cases where hybrid generation fails, and the final rewrite process used to construct the SFT dataset.

Algorithm 1: Student-Friendly Distillation (SFD)

```

1: Input: Off-Policy Dataset  $\mathcal{D} = \{(q, r^*, y^*)\}$ , Student  $\pi_\phi$ , Smoothing Window  $W$ , Stability
   Horizon  $P$ , Thresholds  $\tau_{\text{align}}, \tau_{\text{conf}}$ , The Number of Samples  $N$ 
2: Initialize:  $\mathcal{D}_{\text{SFT}} \leftarrow \emptyset$ 
3:
4: for each  $(q, r^*, y^*) \in \mathcal{D}$  do
5:   // Step 1: Teacher Guidance & Dynamic Hand-off
6:   Let  $r^* = (t_1, \dots, t_T)$ 
7:   Compute token metrics for  $i \in [1, T]$ :
8:    $\mathcal{L}_i \leftarrow -\log \pi_\phi(t_i | q, t_{<i})$ 
9:    $\mathcal{H}_i \leftarrow \mathbb{H}[\pi_\phi(\cdot | q, t_{<i})]$ 
10:  Compute smoothed metrics over window  $W$ :
11:   $S(\mathcal{L})_i \leftarrow \frac{1}{W} \sum_{j=i-W+1}^i \mathcal{L}_j$ 
12:   $S(\mathcal{H})_i \leftarrow \frac{1}{W} \sum_{j=i-W+1}^i \mathcal{H}_j$ 
13:
14:  Define stability indicator  $\delta_i$ :
15:   $\delta_i \leftarrow \mathbb{I}(S(\mathcal{L})_i < \tau_{\text{align}} \vee S(\mathcal{H})_i < \tau_{\text{conf}})$ 
16:  Determine cut-off index  $k$ :
17:   $k \leftarrow \min \left\{ t \in [1, T] \mid \prod_{j=t}^{\min(t+P, T)} \delta_j = 1 \right\}$ 
18:  Set teacher prefix  $p \leftarrow (t_1, \dots, t_k)$ 
19:
20:  // Step 2: Hybrid Teacher-Student Generation
21:   $\mathcal{R}_{\text{hybrid}} \leftarrow \emptyset$ 
22:  for  $n = 1$  to  $N$  do
23:    Sample Completion  $s_n \sim \pi_\phi(\cdot | q, p)$ 
24:    Form candidate  $r_{\text{cand}} \leftarrow p \oplus s_n$ 
25:    if  $\text{Verify}(r_{\text{cand}}, y^*)$  then
26:       $\mathcal{R}_{\text{hybrid}} \leftarrow \mathcal{R}_{\text{hybrid}} \cup \{r_{\text{cand}}\}$ 
27:    end if
28:  end for
29:
30:  // Step 3: Scaffold Selection (Fallback Logic)
31:  if  $\mathcal{R}_{\text{hybrid}} \neq \emptyset$  then
32:     $c_{\text{scaffold}} \leftarrow \text{RandomSelect}(\mathcal{R}_{\text{hybrid}})$ 
33:  else
34:     $c_{\text{scaffold}} \leftarrow r^* \text{ {Fallback: use original teacher trace}}$ 
35:  end if
36:
37:  // Step 4: Student Rewrite (Manifold Projection)
38:   $\mathcal{R}_{\text{final}} \leftarrow \emptyset$ 
39:  for  $n = 1$  to  $N$  do
40:     $r_{\text{rw}} \sim \pi_\phi(\cdot | q, \text{context} = c_{\text{scaffold}})$ 
41:    if  $\text{Verify}(r_{\text{rw}}, y^*)$  then
42:       $\mathcal{R}_{\text{final}} \leftarrow \mathcal{R}_{\text{final}} \cup \{r_{\text{rw}}\}$ 
43:    end if
44:  end for
45:  {Data Accumulation: Add all valid rewrites}
46:  for all  $r_{\text{final}} \in \mathcal{R}_{\text{final}}$  do
47:     $\mathcal{D}_{\text{SFT}} \leftarrow \mathcal{D}_{\text{SFT}} \cup \{(q, r_{\text{final}})\}$ 
48:  end for
49: end for

```

Table 2: Hyperparameters used in **SFD**. The parameters are categorized into Stability Detection (determining the optimal hand-off point k) and Generation (both completion and rewrite).

Category	Parameter	Value	Description
Stability Detection	Window Size (W)	15	Size of the sliding window for metric smoothing.
	Stability Horizon (P)	2	Minimum consecutive tokens required for stability.
	Stride (S)	5	Step size for the sliding window calculation.
	Alignment Threshold (τ_{align})	0.7	Maximum allowed smoothed NLL ($\bar{\ell}_i$).
	Confidence Threshold (τ_{conf})	0.6	Maximum allowed smoothed Entropy (\bar{h}_i).
Generation	Sample Count (N)	8	Number of suffix/rewrite candidates sampled per input.
	Temperature (T)	0.8	Sampling temperature for exploration.
	Top- p	0.8	Nucleus sampling probability mass.
	Max Tokens	16384	Maximum generation length for suffixes.

Table 3: Hyperparameters for the Training (SFT) and Evaluation phases.

Category	Parameter	Value	Description
Training (SFT)	Optimizer	AdamW	Standard optimizer for LLM training.
	Learning Rate	5e-6	Peak learning rate with cosine decay schedule.
	Batch Size	32	Effective batch size (including gradient accumulation).
	Epochs	5	Number of passes over the full dataset.
	Weight Decay	1e-4	Regularization term applied to weights.
	Warmup Ratio	0.1	Percentage of steps used for linear learning rate warmup.
	Max Sequence Length	32768	Maximum context length during training.
Evaluation	Repetitions (N)	4 or 8	The number of generations per prompt for robustness.
	Temperature (T)	1.0	Sampling temperature for evaluation.
	Top- p	0.95	Nucleus sampling probability mass.
	Max Tokens	16384	Maximum generation length for evaluation.

D IMPLEMENTATION DETAILS

D.1 HYPERPARAMETERS OF **SFD**

We detail the hyperparameters used for the dynamic hand-off mechanism and the student generation phases of **SFD** in Table 2. These parameters control the sensitivity of the teacher-student hand-off and the diversity of the candidate generation.

Parameter Roles. Below, we provide the intuition behind the key hyperparameters in **SFD**:

- **Window Size (W) & Stride:** To mitigate token-level volatility in the student’s likelihood estimates, we apply a moving average filter. A window size of $W = 15$ ensures that short-term spikes in perplexity (*e.g.*, due to a single difficult token) do not prematurely trigger or prevent a hand-off. We use a stride of 5 to evaluate stability at regular intervals, which reduces computational overhead while sufficiently capturing the local trend.
- **Stability Horizon (P):** The horizon P enforces a robustness constraint. The student must satisfy the stability criteria ($\delta_i = 1$) for P consecutive steps before the hand-off is finalized. This prevents the algorithm from handing off control during a transient dip in entropy that does not reflect true model confidence.
- **Thresholds ($\tau_{\text{align}}, \tau_{\text{conf}}$):** These thresholds determine the strictness of the hand-off.
 - τ_{align} bounds the negative log-likelihood, ensuring the student’s distribution is not diverging significantly from the teacher’s path.
 - τ_{conf} bounds the entropy, ensuring the student is “certain” about its next token prediction.
- **Sample Count (N):** This parameter balances the trade-off between computational cost and coverage of the solution manifold. We set $N = 8$ for both the Hybrid Sampling and Student Rewrite phases to ensure a sufficiently diverse set of candidates ($\mathcal{R}_{\text{hybrid}}$ and $\mathcal{R}_{\text{final}}$) for the verification oracle.

Hyperparameter Selection (Adaptive Thresholding). To ensure robustness across different queries and model checkpoints, we employ an adaptive thresholding strategy. While we establish fixed upper bounds (listed in Table 2) to serve as safety caps, the effective threshold for each instance is determined dynamically. Specifically, we calculate the 15th percentile of the observed NLL and entropy values across the current batch and set the operating threshold as the minimum of the fixed value and this relative quantile: $\tau = \min(\tau_{\text{fixed}}, \tau_{15\%})$. This ensures that the hand-off requires the student to be in its locally optimal confidence zone (relative to the current context) while maintaining a hard ceiling to prevent hand-offs in uniformly high-uncertainty scenarios. We set P to 2 so that a single rapid decrease in entropy or NLL does not lead to a hand-off.

D.2 HYPERPARAMETERS FOR TRAINING AND EVALUATION

We provide the general settings used for training and the evaluation generation setup in Table 3. For off-policy distillation (SFT in our baseline), we use a different learning rate of $1e-5$ and train for 8 epochs, as this setting requires more training due to distributional mismatch and slower learning speed.

E MORE EXAMPLES

We add additional examples of generated data, along with some qualitative analysis of trained models using ours and off-policy data.

E.1 EXAMPLES OF GENERATED DISTILLATION DATA

The following examples show data generated by **SFD** on Qwen2.5-1.5B and Llama3.2-3B, alongside the student response and the teacher response. Reasoning associated with successful outcomes is highlighted in green, whereas reasoning associated with failures is highlighted in red. As observed, the prefix plays a key role in helping the student complete the correct answer, and it also encourages students to produce more content, making the final response more aligned with the student response.

Example generated data using **SFD** on Qwen2.5-1.5B

Question. Given that $ABC - A_1B_1C_1$ is a right prism with $\angle BAC = 90^\circ$, points D_1 and F_1 are the midpoints of A_1B_1 and B_1C_1 , respectively. If $AB = CA = AA_1$, find the cosine of the angle between BD_1 and CF_1 .

Teacher Response. To solve the problem, we analyze the geometry of the right prism $ABC - A_1B_1C_1$,

...

Since $AB = CA = 1$ and $\angle BAC = 90^\circ$, place: $- B = (1, 0, 0) - C = (0, 1, 0)$
 Since the prism is right and $AA_1 = 1$, the top face is: $- A_1 = (0, 0, 1) - B_1 = (1, 0, 1) - C_1 = (0, 1, 1)$

...

Let D_1 and F_1 be the midpoints of A_1B_1 and B_1C_1 , respectively. Their coordinates are:

$$D_1 = \left(\frac{1}{2}, 0, 1\right), \quad F_1 = \left(\frac{1}{2}, \frac{1}{2}, 1\right)$$

...

Using these points, we form the vectors $\vec{BD_1}$ and $\vec{CF_1}$, compute their dot product and magnitudes, and apply the cosine formula:

$$\cos \theta = \frac{\vec{BD_1} \cdot \vec{CF_1}}{|\vec{BD_1}| |\vec{CF_1}|}$$

...

After simplification, we obtain:

$$\frac{\sqrt{30}}{10}$$

Student Response. To find the cosine of the angle between BD_1 and CF_1 , we analyze the given right prism $ABC - A_1B_1C_1$ with $\angle BAC = 90^\circ$ and $AB = CA = AA_1$. Then $\triangle ABC$ is an isosceles right triangle, and AA_1 is perpendicular to the base.

...

Place the prism in a coordinate system as follows: $A = (0, 0, 0)$, $B = (a, 0, 0)$, $C = (0, a, 0)$, $A_1 = (0, 0, h)$, $B_1 = (a, 0, h)$, $C_1 = (0, a, h)$, where $h = a$.

...

The coordinates of the midpoints are:

$$D_1 = \left(\frac{a}{2}, 0, \frac{h}{2} \right)$$

$$F_1 = \left(\frac{a}{2}, \frac{a}{2}, \frac{h}{2} \right)$$

Using these points, we compute the vectors BD_1 and CF_1 , evaluate their dot product and magnitudes, and obtain:

$$\cos \theta = \frac{h^2 - a^2}{a^2 + h^2}$$

Since $h = a$, we conclude:

$$0$$

Hybrid Response. To solve the problem, we analyze the geometry of the right prism $ABC - A_1B_1C_1$,

...

Since $AB = CA = 1$ and $\angle BAC = 90^\circ$, place: - $B = (1, 0, 0)$ - $C = (0, 1, 0)$

Since the prism is right and $AA_1 = 1$, the top face is: - $A_1 = (0, 0, 1)$ - $B_1 = (1, 0, 1)$ - $C_1 = (0, 1, 1)$

Step 2: Determine Coordinates of Midpoints. D_1 , being the midpoint of A_1B_1 , is

...

Conclusion: the cosine of the angle between BD_1 and CF_1 is:

$$\frac{\sqrt{30}}{10}$$

Rewritten Response. Given a right prism $ABC - A_1B_1C_1$ with $AB = CA = 1$ and $\angle BAC = 90^\circ$, let D_1 and F_1 be the midpoints of A_1B_1 and B_1C_1 , respectively. We aim to find the cosine of the angle between \vec{BD}_1 and \vec{CF}_1 .

Place the prism in a coordinate system by setting $A = (0, 0, 0)$, $B = (1, 0, 0)$, $C = (0, 1, 0)$, and $A_1 = (0, 0, 1)$, $B_1 = (1, 0, 1)$, $C_1 = (0, 1, 1)$. Then

$$D_1 = \left(\frac{1}{2}, 0, 1 \right), \quad F_1 = \left(\frac{1}{2}, \frac{1}{2}, 1 \right).$$

From this, the vectors $B\vec{D}_1$ and $C\vec{F}_1$ can be formed, and their dot product and magnitudes computed. Applying the cosine formula,

$$\cos \theta = \frac{B\vec{D}_1 \cdot C\vec{F}_1}{|B\vec{D}_1| |C\vec{F}_1|} = \frac{\sqrt{30}}{10}.$$

Example generated data using **SFD** on Llama3.2-3B

Question. Two nonhorizontal, nonvertical lines in the xy -coordinate plane intersect to form a 45° angle. One line has slope equal to 6 times the slope of the other line. What is the greatest possible value of the product of the slopes of the two lines?

Teacher Response. To solve the problem, we are given two non-horizontal and non-vertical lines in the xy -plane that intersect at a 45° angle. One line has a slope that is six times the slope of the other. Our goal is to find the **greatest possible value of the product of the two slopes**.

...

Since the lines form a 45° angle, we can use the fact that the tangent of the angle between two lines is equal to the absolute value of the difference of their slopes divided by the product of their slopes. This gives us the equation $|\tan 45^\circ| = \frac{|m-6m|}{|m \times 6m|}$.

$$\Rightarrow \left| \frac{6m - m}{1 + m \cdot 6m} \right| = 1$$

Simplify the expression:

$$\left| \frac{5m}{1 + 6m^2} \right| = 1$$

This gives two cases:

$$1. \frac{5m}{1+6m^2} = 1 \quad 2. \frac{5m}{1+6m^2} = -1$$

...

Thus, the **greatest possible value** of the product is:

$$\frac{3}{2}$$

Student Response. To find the greatest possible value of the product of the slopes of the two lines, let's consider the given information. We have two nonhorizontal, nonvertical lines that intersect to form a 45° angle. The slope of one line is 6 times the slope of the other line. Let's denote the slope of the smaller line as m . Then, the slope of the larger line is $6m$. We want to maximize the product of these slopes, which is $m \times 6m = 6m^2$.

Since the lines form a 45° angle, we can use the fact that the tangent of the angle between two lines is equal to the absolute value of the difference of their slopes divided by the product of their slopes. This gives us the equation $|\tan 45^\circ| = \frac{|m-6m|}{|m \times 6m|}$.

Simplifying, we get $1 = \frac{|5m|}{6|m^2|}$.

...

Therefore, the greatest possible value of the product of the slopes of the two lines is $\frac{25}{6}$.

Hybrid Response. To solve the problem, we are given two non-horizontal and non-vertical

lines in the xy -plane that intersect at a 45° angle. One line has a slope that is six times the slope of the other. Our goal is to find the **greatest possible value of the product of the two slopes**.

...

Let the slopes of the two lines be m and $6m$. The tangent of the angle θ between two lines with slopes m_1 and m_2 is given by:

$$\tan \theta = \left| \frac{m_2 - m_1}{1 + m_1 m_2} \right|$$

Given that the angle is 45° , we have:

$$\tan(45^\circ) = 1 \Rightarrow \left| \frac{6m - m}{1 + m \cdot 6m} \right| = 1$$

Simplify the expression:

$$\left| \frac{5m}{1 + 6m^2} \right| = 1$$

This gives two cases:

$$1. \frac{5m}{1+6m^2} = 1 \quad 2. \frac{5m}{1+6m^2} = -1$$

Cutoff Point

STEP 2: SOLVE THE CASES

Case 1:

$$\frac{5m}{1 + 6m^2} = 1 \Rightarrow 5m = 1 + 6m^2 \Rightarrow 6m^2 - 5m + 1 = 0$$

We can use the quadratic formula to solve for m :

$$m = \frac{5 \pm \sqrt{25 - 24}}{12} = \frac{5 \pm 1}{12}$$

...

Comparing the results, we see that the greatest possible value of the product of the slopes is $3/2$.

$$\boxed{\frac{3}{2}}$$

Rewritten Response.

STEP 1: UNDERSTAND THE PROBLEM

We are given two non-horizontal, non-vertical lines that intersect at a 45° angle. One line has a slope that is six times the slope of the other line. We want to find the greatest possible value of the product of the slopes of the two lines.

...

STEP 3: APPLY THE ANGLE BETWEEN TWO LINES FORMULA

The tangent of the angle between two lines with slopes m_1 and m_2 is given by $\tan \theta = \left| \frac{m_2 - m_1}{1 + m_1 m_2} \right|$. Since the angle between the lines is 45° , we have $\tan(45^\circ) = 1$, and the equation becomes $\left| \frac{6m - m}{1 + m \cdot 6m} \right| = 1$.

STEP 4: SIMPLIFY THE EQUATION

We can simplify the equation to $\left| \frac{5m}{1+6m^2} \right| = 1$.

STEP 5: SOLVE THE ABSOLUTE VALUE EQUATION

This gives us two cases: $\frac{5m}{1+6m^2} = 1$ and $\frac{5m}{1+6m^2} = -1$.

...

STEP 9: DETERMINE THE GREATEST PRODUCT

Comparing the products, the greatest possible value of the product of the slopes is $\frac{3}{2}$.

The final answer is: $\frac{3}{2}$

E.2 GENERATED RESPONSES FROM TRAINED MODELS

The following examples show responses generated by models trained with **SFD** and with SFT, and compare their outputs. The examples are drawn from the MATH500 dataset (Hendrycks et al., 2021). The correct answer is highlighted in green, whereas an incorrect answer is highlighted in red. These examples suggest that, even on simple questions, SFT can suffer from distributional misalignment: the student model memorizes the teacher’s surface-level reasoning patterns rather than learning a correct problem-solving approach, leading to failures even on easy problems. **SFD** addresses this by generating distillation data that lies within the student’s manifold, so models trained with **SFD** produce responses that follow the underlying reasoning and solve the problems correctly.

Comparison of trained Qwen2.5-3B models:
SFT vs. **SFD**

Question. The set of points (x, y, z) that satisfy

$$2x = 3y = -z$$

is a line.

The set of points (x, y, z) that satisfy

$$6x = -y = -4z$$

is another line.

Find the angle between these lines, in degrees.

Response from SFT-trained model. To determine the angle between the two given lines in 3D space, we identify the direction vectors.

...

For the first line, the direction vector is:

$$\vec{v}_1 = \left(\frac{1}{2}, \frac{1}{3}, -1 \right)$$

For the second line, the direction vector is:

$$\vec{v}_2 = (1, -6, -4)$$

...

We claim that the second line is a scalar multiple of the first, $\vec{v}_2 = 3 \cdot \vec{v}_1$, meaning the lines are parallel.

Thus, the angle is 0.

FINAL ANSWER

0

Response from SFD-trained model. To find the distance between $(2, -6)$ and $(-4, 3)$, we use the distance formula:

$$\text{Distance} = \sqrt{(-4 - 2)^2 + (3 - (-6))^2}$$

...

Simplifying the expression, we get:

$$\sqrt{117} = \sqrt{9 \times 13} = 3\sqrt{13}$$

FINAL ANSWER

3√13

Comparison of trained Qwen2.5-1.5B models:
SFT vs. SFD

Question. What is the smallest positive perfect cube that can be written as the sum of three consecutive integers?

Response from SFT-trained model. To determine the angle between the two lines, we derive the direction vectors.

...

Based on the equations, we find the direction vectors to be:

$$\mathbf{v}_1 = \langle 4, 6, -2 \rangle, \quad \mathbf{v}_2 = \langle 1, -6, 1 \rangle$$

...

Using the dot product formula and simplifying, we conclude:

$$\theta \approx \arccos(0.516) \approx 60^\circ$$

FINAL ANSWER

60

Response from SFD-trained model. To find the smallest positive perfect cube that can be written as the sum of three consecutive integers, we note their sum is $3n$.

...

For $3n = N^3$, N must be a multiple of 3. The smallest positive integer is $N = 3$.

...

Thus, the smallest perfect cube is:

$$N^3 = 3^3 = 27$$

FINAL ANSWER

27

Comparison of trained Llama3.2-3B models:
SFT vs. SFD

Question. A regular hexagon can be divided into six equilateral triangles. If the perimeter of one of the triangles is 21 inches, what is the perimeter, in inches, of the regular hexagon?

Response from SFT-trained model. To solve the problem,

...

the perimeter of the hexagon is $6 \times 3s = 18s$.

...

Since the perimeter of one triangle is 21, we have:

$$3s = 21 \implies s = 7$$

...

Thus, the total perimeter is:

$$18s = 18 \times 7 = 126$$

FINAL ANSWER

126

Response from SFD-trained model. To solve the problem, we consider the structure of the hexagon.

...

Given the perimeter of one triangle is 21, we have:

$$3s = 21 \implies s = 7$$

...

Therefore, the perimeter of the regular hexagon is:

$$6 \times 7 = 42$$

FINAL ANSWER

42

LLM USAGE

Every part of the research, including the concepts and ideas, originates entirely from the authors, and all experiments are conducted by ourselves. The authors also wrote the paper manually, though LLMs were used only to check typos and polish minor grammar in some parts.