Skill-Aware Data Selection and Fine-Tuning for Data-Efficient Reasoning Distillation

Lechen Zhang Yunxiang Zhang Wei Hu Lu Wang
University of Michigan, Ann Arbor
{leczhang, yunxiang, vvh, wangluxy}@umich.edu

Abstract

Large reasoning models such as DeepSeek-R1 and their distilled variants achieve strong performance on complex reasoning tasks. Yet, distilling these models often demands large-scale data for supervised fine-tuning (SFT), motivating the pursuit of data-efficient training methods. To address this, we propose a *skill-centric* distillation framework that efficiently transfers reasoning ability to weaker models with two components: (1) **Skill-based data selection**, which prioritizes examples targeting the student model's weaker skills, and (2) **Skill-aware fine-tuning**, which encourages explicit skill decomposition during problem solving. With only 1,000 training examples selected from a 100K teacher-generated corpus, our method surpasses random SFT baselines by +1.6% on Qwen3-4B and +1.4% on Qwen3-8B across five mathematical reasoning benchmarks. Further analysis confirms that these gains concentrate on skills emphasized during training, highlighting the effectiveness of skill-centric training for efficient reasoning distillation.

1 Introduction

Large reasoning models such as DeepSeek-R1 (DeepSeek-AI et al., 2025) achieve impressive performance on complex reasoning tasks, yet their costs remain substantial. Distilling these capabilities into weaker Large Language Models (LLMs) via supervised fine-tuning (SFT) is a promising way to broaden the access. A key challenge, however, is the strategy of choosing the right SFT data. Current pipelines typically treat all training examples equally (DeepSeek-AI et al., 2025), which overlooks the latent structure of data such as the underlying skills, as well as the model's current knowledge state. In contrast, human learning is highly structured, with knowledge organized hierarchically (Gagne, 1962; White, 1973). Motivated by this, we ask whether data selection informed by structured relationships among training examples can similarly improve the learning efficiency of LLMs in reasoning.

Prior work has made various efforts on structured training, notably by formalizing the notion of LLM *skills*, typically defined as "atomic" competencies (e.g., addition or multiplication) when solving problems (Chen et al., 2023; Li et al., 2025a). Some studies (Li et al., 2025a) have explored skill-oriented distillation for LLMs, often by emphasizing broader data coverage (Ye et al., 2025). Yet these approaches have two limitations. First, these approaches usually treat each problem as a single unit (Zeng et al., 2025), ignoring the fact that a single question often involves many atomic skills. Second, prior work (Muennighoff et al., 2025; Ye et al., 2025) generally does not adapt to the model's current strengths and weaknesses. A geometry-proficient model, for example, learns little from redundant geometry data. Our approach rests on a simple principle: **LLMs should be trained more on the atomic skills they struggle with, and less on the ones they already master.**

Another challenge lies in enabling LLMs to grasp the hierarchical structure of skills. Conventional distillation exposes models only to QA pairs, leaving relationships among underlying skills implicit. Prior work (Didolkar et al., 2024) has shown that prompting models with an explicit list of skills can bring improvement. Inspired by this, we inject structured skill information into training data so that models learn to both solve problems and internalize how different levels of skills are organized.

In this work, we introduce a *skill-centric* data construction framework for math reasoning distillation that leverages a hierarchical skill tree (Kaur et al., 2025) to efficiently map examples to multiple relevant skill chains and select targeted data based on per-skill proficiency. Moreover, by embedding interpretable skill chains into the data, the model learns to reason explicitly over a set of skills before answering. Using only 1,000 distillation exam-

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 595–604

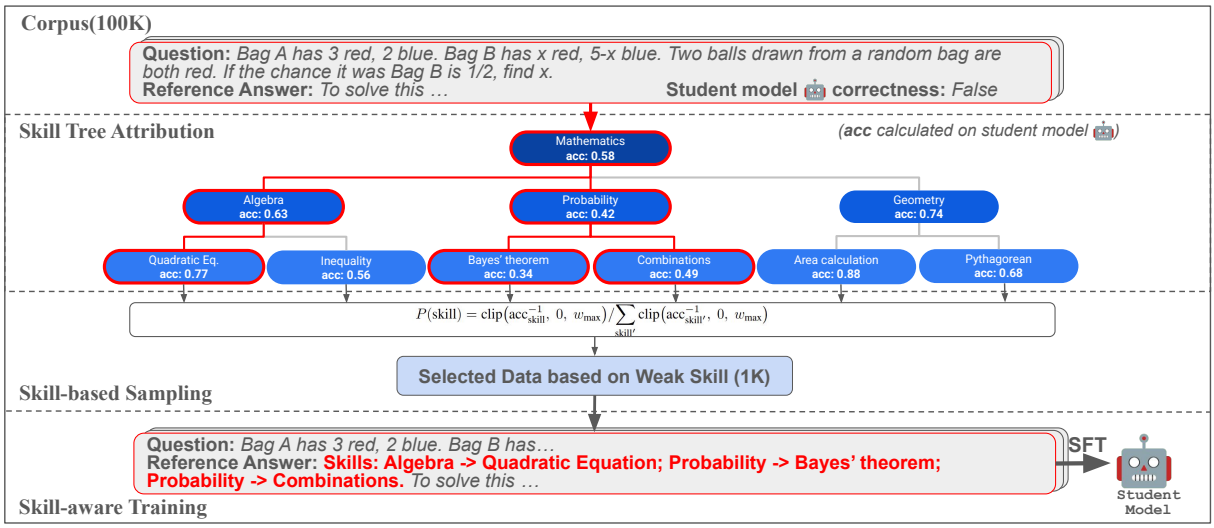

Figure 1: Overview of our skill-centric distillation framework. (1) **Skill Tree Attribution:** Each problem is mapped to nodes on a hierarchical skill tree (Kaur et al., 2025) via top-down LLM-based skill attribution (2) **Skill-based Sampling:** The student model's per-skill accuracy guides sampling, with weaker skills emphasized. (3) **Skill-aware Training:** Selected examples are augmented with explicit skill chains (as shown in red) for skill-aware training.

ples from a 100K corpus, our approach improves Avg@8 accuracy by +1.6% on Qwen3-4B and +1.4% on Qwen3-8B across five math reasoning benchmarks, highlighting the promise of structured, skill-centric distillation for LLM reasoning. The code and dataset are available at https://github.com/orange0629/skill-data-selection.

2 Related Work

Distillation of LLM Reasoning Knowledge distillation was first introduced as a way to compress large neural networks into smaller ones (Hinton et al., 2015), later popularized in NLP (Sanh et al., 2020). Building on these foundations, recent work has shifted toward distilling reasoning abilities. For example, DeepSeek-AI et al. (2025) showed that large models can successfully transfer complex reasoning traces into smaller students with ~800K R1 outputs. Follow-up studies (Moshkov et al., 2025; Yang et al., 2025b; Sun et al., 2025) demonstrate that distilling reasoning can yield compact models that approach much larger systems in reasoning capability. However, these works overlook the impact of data selection in distillation, which we address through skill-aware, model-adaptive training.

Data Selection for Efficient Training Selecting the most informative training examples has long been studied to improve model performance under limited budgets. Recent studies show that small but carefully curated datasets can yield strong reasoning performance. For example, LIMO (Ye et al., 2025), s1 (Muennighoff et al., 2025), and NaturalThoughts (Li et al., 2025b) all demonstrate that high-quality and diverse examples often outperform large-scale random sampling. Nevertheless, existing methods largely ignore structured relations among examples, whereas we leverage a skill hierarchy for fine-grained, interpretable selection.

Skill Decomposition and Structured Reasoning A growing line of work views complex reasoning as a composition of simpler skills and leverages this structure for improved evaluation and training (He et al., 2026). Didolkar et al. (2024) showed that prompting LLMs to identify relevant skills improves math performance. EvalTree (Zeng et al., 2025) organizes tasks into a hierarchical skill tree to locate weak skills for synthesizing targeted data. Instruct-SkillMix (Kaur et al., 2025) combines predefined skills to create instruction data. Rather than synthesizing data with skills, our method integrates skills directly into adaptive data selection.

3 Method

Our approach is motivated by two intuitions: (1) models should receive more training data on skills they are weak at, and (2) models generalize more effectively if they are explicitly trained to recognize skill structures. Our workflow, as shown in Figure 1, begins with a corpus of 100K math QA pairs, and a pre-defined skill tree that categorizes mathematical problems into hierarchical skills.

Base Model	Data Selection	Fine-tuning Strategy	AMC23	AIME2024	AIME2025	MATH L5	OlympiadBench	Average
Qwen3-4B	-	Base	90.1	61.1	50.7	84.3	49.1	67.1
	Full (100K)	Standard SFT	81.9	46.7	34.6	80.2	47.0	58.1
	Random	Standard SFT	89.5	60.1	50.3	85.3	49.0	66.8
	Random	Skill-aware SFT	90.9	62.2	49.9	**85.8**	49.0	67.6
	Skill-based	Standard SFT	89.1	62.5	50.0	85.5	49.5	67.3
	Skill-based	Skill-aware SFT	**91.9**	**64.6**	**50.8**	85.3	**49.6**	**68.4**
Qwen3-8B	-	Base	88.2	61.1	50.2	84.7	49.1	66.7
	Full (100K)	Standard SFT	82.4	47.1	35.5	80.6	46.7	58.5
	Random	Standard SFT	90.2	62.6	50.8	86.0	50.7	68.1
	Random	Skill-aware SFT	91.5	65.7	**52.6**	**86.6**	50.4	69.4
	Skill-based	Standard SFT	**93.4**	62.1	51.3	86.2	49.7	68.5
	Skill-based	Skill-aware SFT	91.9	**67.1**	50.0	**86.6**	**51.6**	**69.5**

Table 1: Accuracy (%) of Qwen3-4B and Qwen3-8B under different training data selection and fine-tuning strategies using **1K training examples**. Each column within each base model block is **bolded** at its highest value and underlined at its second highest. Results are reported using Avg@8 across five math benchmarks. Notably, fine-tuning on the full 100K corpus underperforms the base model, highlighting the importance of data selection.

Step 1: Skill tree attribution Each training problem is mapped onto the tree by attributing its reference solution to relevant skills. Starting from the root, we prompt *Qwen2.5-32B-Instruct* (Qwen Team, 2024) to decide which high-level skill is involved (prompt shown in Appendix B). For each selected skill, the LLM is further asked to drill down the decision at the next level, until the leaf node is reached. This recursive process leverages the hierarchical structure (with $O(\log N)$ complexity) to avoid overwhelming the model with a flat multi-label decision and ensures comprehensive coverage of all required skills.[1]

Step 2: Skill-based sampling To adapt training data to a model's weaknesses, we evaluate the student model on the 100K corpus. For each leaf skill, we compute the model's accuracy, yielding a skill-wise performance profile. Training examples are then sampled with probabilities inversely proportional to these accuracies: $P(\text{skill}) = \frac{\text{clip}\left(\text{acc}_{\text{skill}}^{-1},\, 0,\, w_{\max}\right)}{\sum_{\text{skill}'} \text{clip}\left(\text{acc}_{\text{skill}'}^{-1},\, 0,\, w_{\max}\right)}$, where $w_{\max}$ is empirically set to 10,000 to cap divide-by-zero issue. This ensures that underrepresented or difficult skills are emphasized while preventing excessive redundancy in well-mastered ones. Using this distribution, we construct our training subsets of 1K.

Step 3: Skill-aware training Finally, we prepare skill-aware variants of the training data by embedding the explicit skill chain into each instance. For each problem, the ordered sequence of required skills, e.g., *"Skills: [Mathematics → Probability → Bayes' theorem]"*, is prepended before the solution. This encourages the model to explicitly traverse the required skills before attempting the solution, enabling fine-grained diagnostics of model performance at the skill level.

[1]We manually inspected ∼100 random QA pairs and found no evidence of missing or mislabeled skills.

4 Experiments

We conduct a series of experiments to evaluate the effectiveness of our skill-centric pipeline.

4.1 Setup

We experiment with two reasoning models: **Qwen3-4B** and **Qwen3-8B** (Qwen Team, 2025). We extract a clean set of 100K unique QA pairs from **OpenMathReasoning** (Moshkov et al., 2025) as our teacher data pool (Details in Appendix A). In our experiments, we adopt the existing skill tree hierarchy proposed in the *Instruct-SkillMix* paper (Kaur et al., 2025) to label skills for all data.

All models are fine-tuned for 5 epochs (details in Appendix E). Evaluation is conducted on five diverse competition-style math benchmarks: **AMC23**, **AIME2024**, **AIME2025**, **MATH L5** (Level 5) (Hendrycks et al., 2021), and **OlympiadBench** (He et al., 2024). Avg@8 accuracy (calculated by the average accuracy over 8 independent samples per question) is reported.

4.2 Main Results and Analysis

Table 1 shows the performance across different training strategies. We observe that: **Skill-tree-based data selection generally outperforms random sampling.** For Qwen3-4B, Skill-based data selection yields a +0.5 gain in average accuracy, with the largest improvements on AIME2024 (+2.4). Similarly, Qwen3-8B has a +0.4 average gain with significant improvement on AMC23

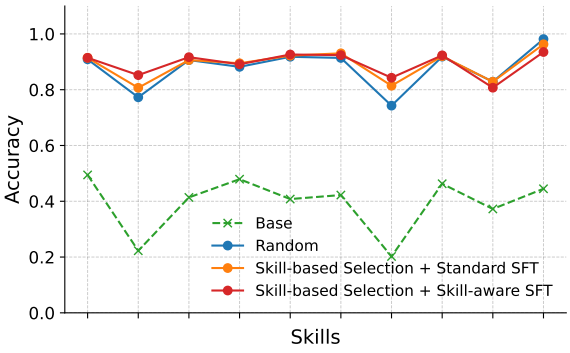

Figure 2: Per-skill accuracy shift of skill-based selection on **MATH-500**. Skill-based sampling improves weaker skills while preserving strong ones, flattening the accuracy curve toward balanced mastery. Skill-aware augmentation further enhances robustness across skills. (Detailed version is in Appendix Figure 3)

(+3.2). These results indicate that aligning training with the model's weaker skills provides consistent benefits to LLMs. Second, **skill-aware training consistently provides additional gains over standard SFT.** Adding explicit skill chains improves average accuracy in nearly all settings, with the largest boost on AIME2024 (up to +5.0), and strongest overall gains of up to +0.8 for Qwen3-4B and +1.3 for Qwen3-8B. Combining Skill-based data selection with skill-aware augmentation further amplifies the effect, yielding significant improvements over random selection (+1.6 for Qwen3-4B and +1.4 for Qwen3-8B) and delivering the strongest overall results. These findings confirm that skill-aware sampling and training are complementary and robust. Notably, fine-tuning on the full 100K corpus consistently degrades performance relative to the base model, a phenomenon also observed in recent work (Ye et al., 2025), highlighting that data selection quality matters more than quantity for effective reasoning distillation.

To examine the effect of our skill-based oversampling strategy on individual skills, we further evaluate 500 problems from the **MATH-500** (Lightman et al., 2024) benchmark. Figure 2 reports the skill-wise accuracies across different settings (each position on the x-axis represents a distinct skill). Both random and skill-based sampling substantially improve accuracy over the base model. **Skill-based oversampling effectively aligns SFT data distribution with model weaknesses.** Weaker skills that are sampled more frequently correspond to larger accuracy gains. Moreover, the accuracy of stronger skills remains high although sampled less frequently, suggesting that random sampling may waste training cost on areas where the model already performs well. Therefore, **skill-based training curve becomes notably flatter, showing that the model achieves more balanced and robust performance across skills**. Adding skill-aware augmentation further strengthens this effect, yielding even greater consistency in skill performance.

4.3 Ablation Studies

Setting	Avg Accuracy
Effect of Sampling Aggressiveness	
$T = 0.5$	70.7
$T = 0.75$	71.3
$\mathbf{T = 1.0}$	71.9
$T = 2.0$	72.0
$T = 3.0$	71.9
Is the Full Skill Chain Necessary?	
Full skill chain	72.9
Root Skills Only	72.2
Leaf Skills Only	72.7

Table 2: Ablations on Sampling Aggressiveness and Hierarchical Skill Chain. Default settings are **bolded**. Full results are in Appendix Table 6.

Effect of Sampling Aggressiveness We examine how the aggressiveness of weakness sampling influences performance by varying the exponent of accuracy (replacing acc^{-1} in the formula with acc^{-T}). As shown in Table 2, performance first rises quickly and then saturates as sampling becomes more aggressive. Thus, setting $T = 1.0$ provides a simple and effective balance.

Is the Full Chain of Skills Necessary? Our skill-aware SFT provides the full hierarchical skill chain. To test its necessity, we ablate the chain and expose only one layer, either top-level or leaf skills. As shown in Table 2, top-level skills yield little benefit, and leaf-level skills improve more but still underperform the full chain. This suggests that training with the complete skill tree structure is beneficial for model learning.

4.4 Generalization Results

Our approach also demonstrates strong generalization across a wider range of settings. Appendix C includes three complementary results: (i) generalization to alternative skill taxonomies and tree structures (EvalTree; Zeng et al., 2025); (ii) generalization to different model families (R1-Distill-Llama-8B; DeepSeek-AI et al., 2025); and (iii) gains over strong data-selection baselines, including LIMO (Ye et al., 2025), s1 (Muennighoff

et al., 2025), Light-R1 (Wen et al., 2025), and Select2Reason (Yang et al., 2025a). These results support the broad applicability of our skill-aware distillation framework.

5 Conclusion

This research demonstrates that skill-based data selection and skill-aware training enable more capable, data-efficient, and interpretable reasoning distillation. By prioritizing examples from weaker skills of the student model and embedding explicit skill structures during SFT, our approach allows smaller models to acquire strong and robust reasoning abilities. These findings highlight the potential of skill-centric training as a general framework for improving distillation efficiency and transparency.

6 Limitations

While our study demonstrates clear benefits of skill-centric training, several limitations remain. First, our approach relies on existing skill trees. Although this structure covers most mathematical domains, it may not perfectly align with the skill decomposition used by the student model. Future work could explore more skill tree variants or automatically learned skill hierarchies. Second, the accuracy-based sampling assumes that per-skill evaluation reliably reflects model competence. However, skill-wise accuracy can be noisy, especially when each skill has limited evaluation data. A more robust estimate, perhaps through uncertainty modeling or multi-task validation, may improve stability in sampling decisions. Finally, we evaluate only two model scales (4B and 8B). Although results are consistent across three different models, further validation on larger sizes can be very helpful to assess generality.

7 Ethical Considerations

This work focuses on improving data efficiency and interpretability in reasoning model distillation and does not involve any human subjects or personally identifiable information. All datasets used, including OpenMathReasoning and benchmark test sets (e.g., AMC23, AIME, MATH, OlympiadBench), consist of publicly available mathematical problems without sensitive content. We emphasize that skill-centric training aims to enhance transparency and interpretability rather than automate human reasoning. Models trained with our framework should be deployed responsibly, with human oversight and clear communication of their limitations.

Acknowledgments

This work is supported by computational resources and services provided by Advanced Research Computing (ARC), a division of Information and Technology Services (ITS) at the University of Michigan, Ann Arbor.

References

Mayee F Chen, Nicholas Roberts, Kush Bhatia, Jue WANG, Ce Zhang, Frederic Sala, and Christopher Re. 2023. Skill-it! a data-driven skills framework for understanding and training language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Aniket Rajiv Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, Michal Valko, Timothy P Lillicrap, Danilo Jimenez Rezende, Yoshua Bengio, Michael Curtis Mozer, and Sanjeev Arora. 2024. Metacognitive capabilities of LLMs: An exploration in mathematical problem solving. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Robert Gagne. 1962. The acquisition of knowledge. *Psychological Review*, 69:355–365.

Shousheng Jia Haosheng Zou, Xiaowei Lv and Xiangzheng Zhang. 2024. 360-llama-factory.

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

Yinghui He, Abhishek Panigrahi, Yong Lin, and Sanjeev Arora. 2026. STAT: Skill-targeted adaptive training. In *The Fourteenth International Conference on Learning Representations*.

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

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

Simran Kaur, Simon Park, Anirudh Goyal, and Sanjeev Arora. 2025. Instruct-skillmix: A powerful pipeline for LLM instruction tuning. In *The Thirteenth International Conference on Learning Representations*.

Jiazheng Li, Lu Yu, Qing Cui, Zhiqiang Zhang, JUN ZHOU, Yanfang Ye, and Chuxu Zhang. 2025a. MASS: Mathematical data selection via skill graphs for pretraining large language models. In *Forty-second International Conference on Machine Learning*.

Yang Li, Youssef Emad, Karthik Padthe, Jack Lanchantin, Weizhe Yuan, Thao Nguyen, Jason Weston, Shang-Wen Li, Dong Wang, Ilia Kulikov, and Xian Li. 2025b. Naturalthoughts: Selecting and distilling reasoning traces for general reasoning tasks. *Preprint*, arXiv:2507.01921.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. 2025. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. *arXiv preprint arXiv:2504.16891*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *Preprint*, arXiv:2501.19393.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Qwen Team. 2025. Qwen3: Think Deeper, Act Faster | Qwen.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Preprint*, arXiv:1910.01108.

Lin Sun, Guangxiang Zhao, Xiaoqi Jian, Yuhan Wu, Weihong Lin, Yongfu Zhu, Change Jia, Linglin Zhang, Jinzhu Wu, Junfeng Ran, Sai er Hu, Zihan Jiang, Junting Zhou, Wenrui Liu, Bin Cui, Tong Yang, and Xiangzheng Zhang. 2025. Tinyr1-32b-preview: Boosting accuracy with branch-merge distillation. *Preprint*, arXiv:2503.04872.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. 2025. Light-r1: Curriculum SFT, DPO and RL for long COT from scratch and beyond. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 318–327, Vienna, Austria. Association for Computational Linguistics.

Richard T. White. 1973. Research into learning hierarchies. *Review of Educational Research*, 43(3):361–375.

Cehao Yang, Xueyuan Lin, Xiaojun Wu, Chengjin Xu, Xuhui Jiang, Honghao Liu, Hui Xiong, and Jian Guo. 2025a. Select2reason: Efficient instruction-tuning data selection for long-cot reasoning. *Preprint*, arXiv:2505.17266.

Zheyuan Yang, Lyuhao Chen, Arman Cohan, and Yilun Zhao. 2025b. Table-r1: Inference-time scaling for table reasoning. *Preprint*, arXiv:2505.23621.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*.

Zhiyuan Zeng, Yizhong Wang, Hannaneh Hajishirzi, and Pang Wei Koh. 2025. Evaltree: Profiling language model weaknesses via hierarchical capability trees. In *Second Conference on Language Modeling*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

A Data Filtering Details

We use **OpenMathReasoning** (Moshkov et al., 2025) as the primary dataset, a large math reasoning corpus containing 306K unique problems with 3.2M solutions sampled from DeepSeek-R1 (DeepSeek-AI et al., 2025). From the corpus, we construct a 100K clean training pool by applying several filtering steps. First, we discard problems without a ground-truth answer. Each unique problem is associated with approximately ten candidate responses; we retain only those generated by DeepSeek-R1 (DeepSeek-AI et al., 2025) and only when the predicted final answer exactly matches the ground truth. For each problem, we then keep a single valid response to avoid duplication. This procedure yields roughly 105K problem–solution pairs. We ensure that there is no data leakage between our training corpus and the evaluation benchmarks. Finally, we randomly remove 5K instances to obtain a balanced set of 100K unique QA pairs used in our experiments.

B Skill Tree Attribution Details

The prompt we used on *Qwen/Qwen2.5-32B-Instruct* (Qwen Team, 2024) for top-down skill attribution is listed below:

```
Given the following Math problem:

Q&A: {qa_input}

Which of the following skills are
    involved to understanding or
    solving the problem? Even the most
     basic skills such as simple
    addition and subtraction must be
    taken into account. You can select
     multiple options if needed. Just
    return a list of skill names.

Skills:
{chr(10).join([f"- {name}" for name in
     child_names])}

Answer as a Python list of strings.
'''
```

If using a flatter structure, the sampling is still expected to be the same (since sampling is performed at the leaf level). However, flattening introduces substantial challenges for reliable skill attribution and scalability. Our top-down tree attribution operates in O(log N) time by traversing the hierarchy, allowing efficient selection even with large numbers of skills. In contrast, a flat structure requires selecting directly from all N skills simultaneously, which becomes increasingly error-prone and non-reproducible as N grows (e.g., when thousands of math skills are present together).

Also, the hierarchical structure improves the skill-aware augmented SFT. As shown in Table 2, exposing the full hierarchy in the prompt consistently outperforms using only leaf-level skills, demonstrating that hierarchical organization provides useful structure for learning beyond flat skill lists.

C Generalization Results and Additional Comparisons

This section reports additional experiments omitted from the main paper due to space limits. We study (1) sensitivity to the choice of the skill tree, (2) transfer to another model family, and (3) comparisons with existing data selection methods.

C.1 Generalization to Different Skill Tree Design

We evaluate whether our framework depends on the specific choice of skill tree. In the main paper we use Instruct-SkillMix (Kaur et al., 2025). Here we additionally test EvalTree (Zeng et al., 2025), whose hierarchy can be substantially deeper. Table 3 shows that EvalTree yields comparable or better results than Instruct-SkillMix across benchmarks, indicating that the method transfers across substantially different tree designs.

C.2 Generalization to Another Model Family

To test generality beyond Qwen3, we run the same pipeline on *deepseek-ai/DeepSeek-R1-Distill-Llama-8B* (DeepSeek-AI et al., 2025). Table 4 shows the same pattern as Qwen3: skill-based selection improves over random, and adding skill-aware SFT yields additional gains.

C.3 Comparison with Existing Data Selection Methods

While improvements can be numerically small, improving strong reasoning models with only 1,000 SFT examples is inherently challenging. To contextualize this setting, Table 5 compares our method with LIMO (Ye et al., 2025), s1 (Muennighoff et al., 2025), Light-R1 (Wen et al., 2025), and Select2Reason (Yang et al., 2025a) on Qwen3-4B. We evaluate LIMO using its released dataset (offline), since code is not released. For s1, we include both the released dataset (offline) and rerunning their selection code on our 100K pool (online). For Light-R1 and Select2Reason, we selected the data

Base Model	Data Selection	AMC23	AIME2024	AIME2025	MATH L5	OlympiadBench	Average
Qwen3-4B	Base	90.1	61.1	50.7	84.3	49.1	67.1
	Random	89.5	60.1	50.3	85.3	49.0	66.8
	Instruct-SkillMix	89.1	**62.5**	50.0	**85.5**	**49.5**	67.3
	EvalTree	**90.3**	**62.5**	**53.3**	85.4	49.4	**68.2**
Qwen3-8B	Base	88.2	61.1	50.2	84.7	49.1	66.7
	Random	90.2	62.6	50.8	86.0	**50.7**	68.1
	Instruct-SkillMix	**93.4**	62.1	**51.3**	**86.2**	49.7	68.5
	EvalTree	91.3	**65.4**	**51.3**	85.2	50.6	**68.8**

Table 3: Accuracy (%) of Qwen3-4B and Qwen3-8B under different skill taxonomies and tree structures (comparing EvalTree (Zeng et al., 2025) versus Instruct-SkillMix (Kaur et al., 2025)) using 1K training examples. Each column within each base model block is **bolded** at its highest value. The result indicates that our skill-aware method is robust to different skill taxonomies and tree structure.

Base Model	Data Selection	Fine-tuning Strategy	AMC23	AIME2024	AIME2025	MATH L5	OlympiadBench	Average
R1-Distill-Llama-8B	-	Base	79.3	37.1	27.8	61.5	38.5	48.8
	Full (100K)	Standard SFT	74.8	30.9	21.7	59.0	36.2	44.5
	Random	Standard SFT	81.0	36.1	30.4	70.3	**42.4**	52.0
	Random	Skill-aware SFT	82.0	**37.8**	30.9	71.2	42.3	52.8
	Skill-based	Standard SFT	81.3	35.0	31.0	**71.9**	42.2	52.3
	Skill-based	Skill-aware SFT	**83.1**	36.3	**31.3**	71.8	42.3	**53.0**

Table 4: Accuracy (%) of DeepSeek-R1-Distill-Llama-8B under different training data selection and fine-tuning strategies using **1K training examples**. Each column is **bolded** at its highest value and underlined at its second highest. Results are reported using Avg@8 across five math benchmarks.

Method	AMC23	AIME2024	AIME2025	MATH L5	Average
Base	90.1	61.1	50.7	84.3	71.6
Random	89.5	60.1	50.3	85.3	71.3
LIMO (offline)	88.4	**61.3**	45.4	85.3	70.1
s1 (offline)	89.1	57.9	47.5	**85.9**	70.1
s1 (online)	89.1	60.7	50.2	84.9	71.2
Light-R1 (online)	89.7	62.1	51.7	83.3	71.7
Select2Reason (online)	90.7	63.5	50.2	84.8	72.3
Our method	**91.9**	**64.6**	**50.8**	85.3	**73.2**

Table 5: Comparison with existing data selection methods on Qwen3-4B under our 1K SFT setting. Offline uses released datasets, online reruns the selection code on our 100K pool.

based on their method (online). Existing selections degrade performance or match random sampling, whereas our method consistently improves across benchmarks.

D Compute Cost

We report the compute cost of the two main stages in our pipeline: (i) one-time skill labeling (Skill Tree Attribution) over the 100K training pool, and (ii) SFT on the selected 1K subset.

One-time skill labeling (inference-only). Skill attribution is performed by prompting Qwen2.5-32B-Instruct to assign each example to a skill chain via top-down traversal (Appendix B). This stage is *inference-only* and is *model-agnostic* with respect to the student: the labels depend only on the skill definitions and the data, not on the target model being fine-tuned. In our implementation, labeling the 100K pool took approximately **200 GPU hours** in total. Importantly, this labeling cost is *amortizable*: once produced, the same labels can be reused across multiple student models, seeds, and future experiments under the same skill taxonomy. We note that a comparable corpus-level inference cost is common in recent data selection work, which often relies on LLMs to generate auxiliary metadata (e.g., quality, difficulty, topic, or other annotations) for a large candidate pool before selecting a small training subset (Ye et al., 2025; Muennighoff et al., 2025; Li et al., 2025a).

SFT cost. For SFT, our default setting fine-tunes each student model for 5 epochs (Appendix E). Under our 1K-example setting, each training run costs about **40 GPU hours** per run. For the **full 100K** standard SFT baseline, the training cost is substantially higher; in our setup it is approximately **512 GPU hours**. This comparison highlights that, even when accounting for the one-time labeling cost, our pipeline remains compute-efficient in repeated use cases (e.g., evaluating multiple students or running multiple ablations), because the labeling step does not scale with the number of students.

E Training Details

Environment. All experiments were conducted using NVIDIA A40 GPUs with 48GB memory. The software environment was configured as follows:

- `360-LLaMA-Factory` (Haosheng Zou and Zhang, 2024) (A long-CoT adapted version of `LLaMA-Factory` 0.9.1 (Zheng et al., 2024))
- `torch` 2.7.0
- `transformers` 4.51.3
- `accelerate` 1.0.1
- `datasets` 3.1.0
- `trl` 0.9.6
- `peft` 0.12.0
- `deepspeed` 0.14.4

SFT Training. For SFT training, we used the following settings:

- Batch size: 32 (8 GPUs * 4 Gradient Accumulation)
- Epoch: 5
- Learning rate: 1e-5
- Optimizer: AdamW
- Learning rate scheduler: cosine with warmup
- Warmup ratio: 0.1
- Cutoff length: 8192
- Time Cost: 4 hours per run

Decoding Setup. During inference, we applied the following decoding settings:

- Temperature: 0.6
- Max tokens: 16384
- Top-p: 0.95

F Additional Experiment Results

Detailed version of Figure 2 with data proportion shift is shown in Figure 3.

Simple version of the Ablation Study in Section 4.3 is shown in Table 2, and its full version is shown in Table 6.

G Use of Large Language Models

We acknowledge that we only used LLMs to check grammatical errors in the paper and to improve the clarity of expression.

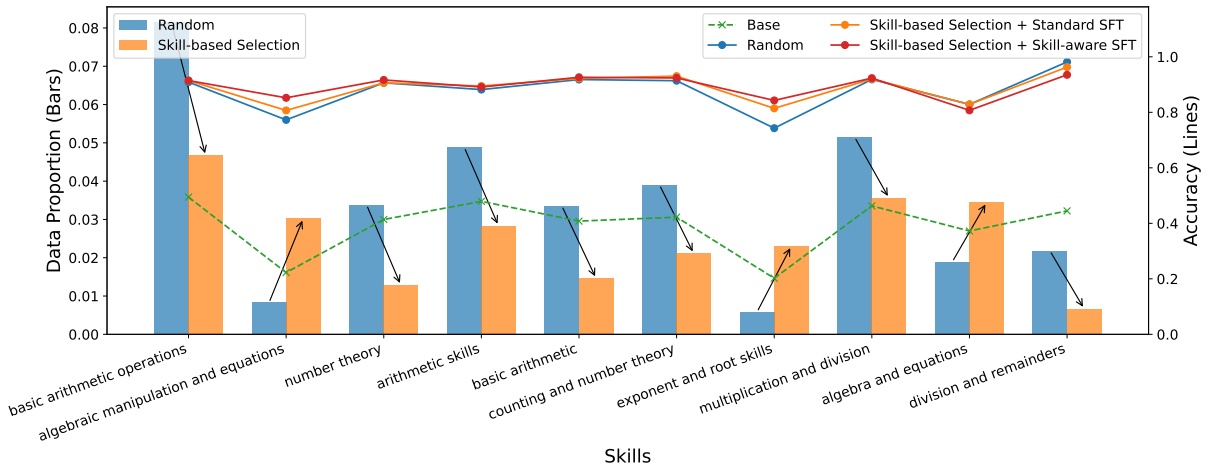

Figure 3: Data proportion shift of skill-based selection and per-skill accuracy (%) on **MATH-500**. Skill-based sampling improves weaker skills while preserving strong ones, flattening the accuracy curve toward balanced mastery. Skill-aware augmentation further enhances robustness across skills.

Ablation	Setting	AMC23	AIME2024	AIME2025	MATH L5	Average
Effect of Sampling Aggressiveness	$T = 0.5$	89.7	60.0	47.9	85.2	70.7
	$T = 1.0$	89.1	**62.5**	**50.0**	<u>85.7</u>	<u>71.9</u>
	$T = 2.0$	<u>90.6</u>	**62.5**	<u>48.8</u>	**85.9**	**72.0**
	$T = 3.0$	**91.6**	<u>61.7</u>	<u>48.8</u>	85.6	<u>71.9</u>
Is the Full Skill Chain Necessary?	Full skill chain	**91.9**	**64.2**	<u>50.4</u>	85.1	**72.9**
	Root Skills Only	<u>91.6</u>	58.3	**52.5**	<u>86.3</u>	72.2
	Leaf Skills Only	90.9	<u>62.9</u>	50.0	**86.9**	<u>72.7</u>

Table 6: Ablations on sampling aggressiveness (T) and on exposing different portions of the skill hierarchy during skill-aware SFT. Within each ablation block, the highest value per column is **bolded** and the second-highest is <u>underlined</u>.