Alibaba Cloud 2026-01-14

# Distribution-Aligned Sequence Distillation for Superior Long-CoT Reasoning

Shaotian Yan[*], Kaiyuan Liu[*], Chen Shen[*,†], Bing Wang[*], Sinan Fan[*], Jun Zhang, Yue Wu, Zheng Wang, Jieping Ye

**Alibaba Cloud Computing**

🤗 Models & Datasets Models & Datasets Code

## Abstract

In this report, we introduce **DASD-4B-Thinking**, a lightweight yet highly capable, fully open-source reasoning model. It achieves state-of-the-art performance among open-source models of comparable scale across challenging benchmarks in mathematics, scientific reasoning, and code generation—even outperforming several larger models (e.g., 32B-scale). We begin by critically reexamining a widely adopted distillation paradigm in the community: supervised fine-tuning (SFT) on teacher-generated responses, also known as sequence-level distillation. Although a series of recent works following this scheme have demonstrated remarkable efficiency and strong empirical performance, they are primarily grounded in the SFT perspective. Consequently, these approaches focus predominantly on designing heuristic rules for SFT data filtering, while largely overlooking the core principle of distillation itself—enabling the student model to learn the teacher's full output distribution so as to inherit its generalization capability. Specifically, we identify three critical limitations in current practice: i) *Inadequate representation of the teacher's sequence-level distribution*; ii) *Misalignment between the teacher's output distribution and the student's learning capacity*; and iii) *Exposure bias arising from teacher-forced training versus autoregressive inference*. In summary, these shortcomings reflect a systemic absence of explicit teacher–student interaction throughout the distillation process, leaving the essence of distillation underexploited. To address these issues, we propose several methodological innovations that collectively form an enhanced sequence-level distillation training pipeline. Remarkably, DASD-4B-Thinking obtains competitive results using only 448K training samples—an order of magnitude fewer than those employed by most existing open-source efforts. To support community research, we publicly release our models and the training dataset.

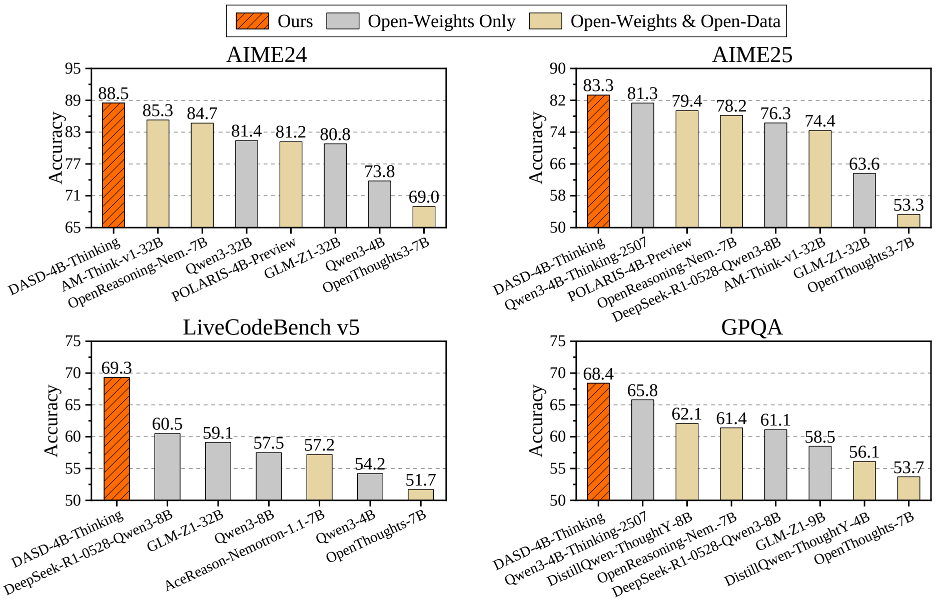

Figure 1: Performance of DASD-4B-Thinking on benchmark datasets. All metrics for the comparison models are taken from their official reports.

[*] Core contributors. [†] Project lead.

## 1 Introduction

Recently, DeepSeek (Guo et al., 2025) was the first to demonstrate that distillation from powerful teacher models can substantially empower smaller models with reasoning capabilities. Specifically, they curated the reasoning data generated by DeepSeek-R1 and directly fine-tuned several widely used open-source compact models. Owing to the simplicity of this supervised fine-tuning (SFT) approach combined with the favorable deployment and inference efficiency of small models, this work has significantly reinvigorated the community's interest in exploring distillation-based methods for reasoning enhancement.

A series of open-source projects (e.g., OpenR1 (Hugging Face, 2025), OpenThoughts (Guha et al., 2025), a-m-team (Zhao et al., 2025), NVIDIA AceReason (Chen et al., 2025b), NVIDIA OpenMathReasoning (Moshkov et al., 2025), OmniThought (Cai et al., 2025), Light-R1 (Wen et al., 2025), LIMO (Ye et al., 2025), s1 (Muennighoff et al., 2025), DeepMath (He et al., 2025), MiroMind-M1 (Li et al., 2025b), Syntheic-1 (Mattern et al., 2025), NaturalThoughts (Li et al., 2025c), and Sky-T1 (Team, 2025)) have since replicated the DeepSeek-R1 distillation paradigm, dedicating substantial efforts to its faithful reproduction and extension. These efforts typically involve collecting and open-sourcing large-scale corpora of challenging reasoning questions, paired with responses generated by powerful teacher models. These datasets have undergone rigorous quality filtering, including stringent correctness verification (Lei et al., 2025; Wu et al., 2025), preference-based selection prioritizing higher reasoning difficulty or longer output length (Muennighoff et al., 2025; Li et al., 2025c), and diversity-aware curation (Jung et al., 2025; Li et al., 2025a). Subsequently, through SFT on such publicly released reasoning corpora, researchers have obtained distilled models that exhibit strong reasoning capabilities. This paradigm of **SFT on teacher-generated responses (Agarwal et al., 2024), i.e., sequence-level distillation (Kim & Rush, 2016)**, has achieved state-of-the-art or highly competitive performance across diverse domains, including mathematics (Hugging Face, 2025; Chen et al., 2025b; Wen et al., 2025; He et al., 2025; Li et al., 2025b), scientific reasoning (Guha et al., 2025; Li et al., 2025c; Mattern et al., 2025), code generation (Ahmad et al., 2025; Guha et al., 2025; Zhao et al., 2025; Cai et al., 2025), and instruction-following (Bercovich et al., 2025; Zhao et al., 2025).

Another paradigm is logit distillation, a classic approach in knowledge distillation (Hinton et al., 2015) that aligns the logit distributions of student and teacher models to better leverage the rich "dark knowledge" encoded in the teacher's outputs. Notably, recent advancements such as Qwen3 (Yang et al., 2025) and Gemma (Kamath et al., 2025) adopt an on-policy variant: they first generate on-policy sequences using the student model and then align the student's logit distributions with those of the teacher by minimizing the KL divergence. Recently, Thinking Machines Lab (Lu & Lab, 2025) released an open-source implementation of this paradigm. However, beyond the requirement of accessing token-level logits, these methods face significant challenges when the teacher and student employ different tokenizers, as direct logit alignment becomes infeasible due to misaligned output spaces.

In this work, we aim to improve the aforementioned paradigm—namely, the sequence-level distillation—**as it is simple and efficient, has already inspired substantial community efforts (including the release of a large number of open-source datasets), imposes no arbitrary constraints on the choice of teacher and student model architectures, and does not require access to token-level logits.** Consistent with Kim & Rush (2016), we first argue that SFT on teacher-generated data serves as an effective form of distillation, since such data approximately reflects the teacher model's sequence-level output distribution and thereby aligns the student model with it. However, **existing works in this first paradigm are primarily grounded in the SFT perspective; consequently, they focus predominantly on designing heuristic rules to filter SFT data, while largely overlooking the core principle of distillation itself—enabling the student model to learn the teacher's full output distribution so as to inherit its generalization capability** (Hinton et al., 2015). In other words, they lack an explicit mechanism to enforce teacher–student interaction throughout the distillation process, leaving the essence of distillation underexploited. More concretely, such approaches neglect three critical issues:

- *From the teacher's perspective: How to better capture and represent the teacher's sequence-level distribution?*

  Existing works randomly sample response data under certain quality-based filtering rules. Although such responses can serve as an approximation of the teacher's sequence-level distribution, this strategy often fails to adequately cover the full support of that distribution. As a result, it may suffer from poor mode coverage or overrepresent low-probability or noisy sequences—making learning particularly challenging for smaller or less capable student models.

- *From the student's learning perspective: How to address misleading gradients when training on teacher-generated data, and more broadly, what target sequence-level distribution better supports effective learning?*

  Classical knowledge distillation leverages the teacher's logit distributions to accurately match the student's predictive distributions over the entire vocabulary at every decoding step, adjusting probabilities up or down as appropriate. In contrast, SFT primarily increases the likelihood of the ground-truth tokens at each prediction position, which can yield misleading gradients (e.g., for tokens the teacher assigns low probabilities but the student assigns high probabilities, SFT pushes those probabilities even higher, driving the student away from the teacher's distribution). Therefore, identifying a teacher sequence-level distribution that is better aligned with the student model's learning is of critical importance.

- *How to mitigate exposure bias caused by training with teacher forcing while evaluating in a free-running, real-world setting?*

  We emphasize that models distilled on teacher-generated data suffer from pronounced exposure bias: during training, they are exposed to teacher-forced inputs, whereas at inference, they must rely entirely on their own autoregressive predictions. This training–inference mismatch induces a distributional shift, leading to error accumulation and compounding deviations over time. Indeed, we observe that the trained student model's outputs often diverge from the training distribution in critical aspects—such as response length—and may enter unexpected states that result in incorrect answers.

We present a preliminary exploration of approaches to address the above limitations and introduce several key advancements to enhance sequence-level distillation for long chain-of-thought (CoT) reasoning:

- **Temperature-scheduled Learning: Broadening coverage of the teacher's modes.** A natural approach is to use a higher sampling temperature to better cover the teacher's output distribution. However, empirical comparison of training convergence across different temperature settings reveals that low-temperature samples exhibit more consistent patterns, making them easier for the student to learn, whereas high-temperature samples—though covering more of the teacher's modes—introduce greater diversity, hindering learning efficiency. This motivates a temperature-scheduled learning strategy: the student first trains on low-temperature, high-confidence samples to grasp consistent patterns, then gradually incorporates higher-temperature samples to broaden mode coverage. Across the multi-domain settings we evaluated, this two-stage training approach achieves performance gains—particularly in complex reasoning domains such as mathematics and code generation—over single-stage training using either high- or low-temperature sampling.
- **Divergence-aware Sampling: Finding target sequence-level distribution better supports effective learning.** Identifying a target distribution—namely, the sequence-level distribution over full responses, as opposed to the token-level logit distribution at each output position—that facilitates effective learning for the student model is nontrivial. Prior work often relies on heuristic rules to select human-expected target distributions, which introduce substantial manual intervention bias and lack theoretical guarantees. To address this, we propose a systematic distribution decomposition framework that analyzes discrepancies between teacher and student predictive probabilities across response candidates. This analysis reveals four canonical distribution patterns. Crucially, we find that one particular pattern—where the teacher assigns high confidence while the student has low probability—consistently correlates with improved test-set performance. Motivated by this finding, we encourage the student model to prioritize learning from such high-divergence instances. Notably, this distribution type naturally mitigates misleading gradients (e.g., from overconfident but incorrect student predictions), and thereby promoting more robust and efficient learning.
- **Mixed-policy Distillation: Mitigating exposure bias of distilled model.** To mitigate exposure bias, we further introduce a lightweight constructively mixed-policy[1] distillation after the initial off-policy SFT phase. Specifically, we randomly select a small subset of training examples, prompt the trained student model to generate full responses, randomly truncate the generated prefixes, and then have the teacher complete the sequence from the truncation point. Only teacher continuations that pass predefined quality filters are retained for the student's fine-tuning. With just a small amount of data and a few additional training steps, this approach yields further performance gains while encouraging more concise model outputs.

Building on these innovations, we present **DASD-4B-Thinking**, a lightweight yet highly capable reasoning model, post-trained via our **D**istribution-**A**ligned **S**equence **D**istillation pipeline. Specifically, we use Qwen3-4B-Instruct-2507 (Yang et al., 2025) as the student model and gpt-oss-120b (Agarwal et al., 2025) as the teacher model, highlighting the broad compatibility of our approach across diverse model families and architectures. Despite substantial differences between the two models in scale, architecture, vocabulary, tokenizer, and pretraining corpora, our pipeline achieves robust distillation performance. We

[1] Mixed-policy refers to data generation involving both student and teacher models.

curate a multi-domain dataset spanning mathematics, code generation, scientific reasoning, and complex instruction-following tasks. Figure 2 illustrates the overall training pipeline. We first sample a small amount of cross-domain data at a low temperature and perform one-stage SFT on the student; we then increase the sampling temperature to generate a larger, more diverse training set, resuming training from the checkpoint of the previous stage. Throughout both stages, all synthetic data are generated using our divergence-aware sampling strategy, designed to better align the teacher's output distribution with the student's learning capacity. Beyond our core innovations, we also inherit established practices from prior work in this line: our pipeline incorporates rigorous quality control measures, such as filtering truncated outputs and repetitive content, to avoid introducing undesirable patterns into the student model. After these stages, the student exhibits strong reasoning capabilities. To further mitigate exposure bias during autoregressive generation, we introduce a lightweight mixed-policy distillation stage, combining teacher-forced and student-sampled trajectories. This yields the final DASD-4B-Thinking, which balances fidelity to high-quality reasoning paths with robustness against self-generated errors.

DASD-4B-Thinking achieves state-of-the-art performance among models of comparable scale across multiple mainstream reasoning benchmarks in mathematics, code generation, and scientific reasoning—even outperforming some larger models (e.g., 32B-scale). Specifically, it attains 88.5 and 83.3 on the highly challenging mathematical competition benchmarks AIME24 and AIME25, respectively; scores 69.3 on LiveCodeBench v5, a widely adopted benchmark for code generation; and achieves 68.4 on GPQA-Diamond, a doctoral-level scientific reasoning benchmark. Notably, thanks to our methodological innovations, these results are obtained using only 448K training samples—an order of magnitude fewer than those employed by most existing open-source efforts.

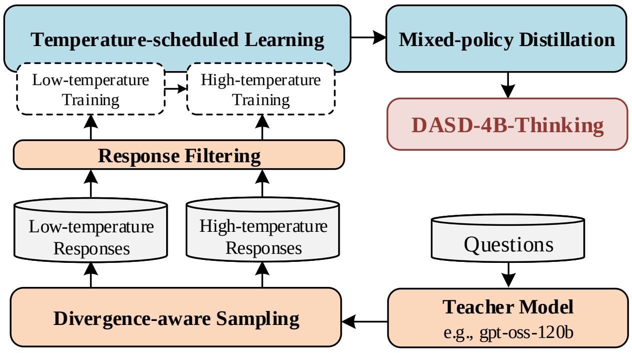
Temperature-scheduled Learning
Low-temperature Training
High-temperature Training
Mixed-policy Distillation
DASD-4B-Thinking
Response Filtering
Low-temperature Responses
High-temperature Responses
Questions
Divergence-aware Sampling
Teacher Model
e.g., gpt-oss-120b

Figure 2: Overall training pipeline of DASD-4B-Thinking.

We have open-sourced our models (DASD-4B-Thinking and a Mixture-of-Experts (MoE) version DASD-30B-A3B-Thinking-Preview—both derivative models of the Qwen family), together with the training dataset on Hugging Face and ModelScope.

In the following sections, we first elaborate on the motivation, methodology, and roles of the three core components of our approach. Section 6 details the full training pipeline implementation, and Section 7 presents a comprehensive evaluation of the models.

## 2 Preliminaries

The current long CoT distillation paradigm of SFT on teacher-generated data can be traced back to the sequence-level distillation (Kim & Rush, 2016), with the original aim of allowing the student model to mimic the teacher's distribution at the sequence level and thereby acquiring capabilities comparable to the teacher. Given an input $\boldsymbol{x}$, it trains a student model $p_S$ to minimize the sequence level divergence from the teacher model $p_T$:

$$minD_{KL}(p_T(\boldsymbol{y} \in \mathcal{Y}|\boldsymbol{x})||p_S(\boldsymbol{y} \in \mathcal{Y}|\boldsymbol{x})), \tag{1}$$

where $\mathcal{Y}$ is the set of all possible responses that the teacher model can generate for the input prompt $\boldsymbol{x}$.

This formulation closely parallels classical logit-based distillation (Hinton et al., 2015); the key difference is that logit distillation typically matches the teacher's conditional next-token distribution at each position, whereas sequence-level distillation aligns with the teacher's distribution over entire output sequences. By modeling the teacher's distribution, logit distillation conveys the teacher's implicit "dark knowledge", enabling the student to inherit its generalization. Analogously, sequence-level distillation pursues the same objective at the level of complete sequences.

Expanding the KL divergence in Equation 1 yields the following sequence-level distillation objective:

$$\mathcal{L}_{SEQ} = \sum_{\boldsymbol{y} \in \mathcal{Y}} p_T(\boldsymbol{y}|\boldsymbol{x})[\log p_T(\boldsymbol{y}|\boldsymbol{x}) - \log p_S(\boldsymbol{y}|\boldsymbol{x})], \tag{2}$$

the value of $p_T(\boldsymbol{y}|\boldsymbol{x}) \log p_T(\boldsymbol{y}|\boldsymbol{x})$ depends only on the teacher and is therefore constant with respect to the student's parameters. It does not affect the gradients and can thus be dropped. Consequently, the sequence-level distillation objective simplifies to:

$$\mathcal{L}_{SEQ} = - \sum_{\boldsymbol{y} \in \mathcal{Y}} p_T(\boldsymbol{y}|\boldsymbol{x}) \log p_S(\boldsymbol{y}|\boldsymbol{x}). \tag{3}$$

However, the exponential size of $\mathcal{Y}$ makes exact computation of $\mathcal{L}_{SEQ}$ intractable. A practical approximation is to use a sampled response $\hat{\boldsymbol{y}}$ and replace $p_T(\cdot \mid \boldsymbol{x})$ with a point mass at $\hat{\boldsymbol{y}}$:

$$p_T(\boldsymbol{y} \mid \boldsymbol{x}) \approx \mathbb{1}\{\boldsymbol{y} = \hat{\boldsymbol{y}}\}. \tag{4}$$

Kim & Rush (2016) adopt beam search to produce $\hat{\boldsymbol{y}}$, which approximates the mode of $p_T(\cdot|\boldsymbol{x})$ (i.e., the response with the highest probability). In contrast, much of the recent literature on long CoT distillation employs randomly sampled responses as $\hat{\boldsymbol{y}}$. After this approximation, $\mathcal{L}_{SEQ}$ can be further simplified to:

$$\mathcal{L}_{SEQ} \sim - \sum_{\boldsymbol{y} \in \mathcal{Y}} \mathbb{1}\{\boldsymbol{y} = \hat{\boldsymbol{y}}\} \log p_S(\boldsymbol{y}|\boldsymbol{x}) = - \log p_S(\hat{\boldsymbol{y}}|\boldsymbol{x}), \tag{5}$$

which exactly recovers the standard SFT loss on teacher-generated outputs. **This perspective clarifies that the success of SFT on teacher-generated data hinges on effectively transferring the teacher's sequence-level distribution to the student model**.

Building on the above analysis, **we argue that the current sequence-level long CoT distillation paradigm should place greater emphasis on teacher–student interaction throughout the distillation process**. However, most existing methods are primarily grounded in the SFT perspective. Consequently, they prioritize filtering high-quality teacher outputs (i.e., SFT data) while neglecting such interaction, leading to three main limitations: (i) Inadequate coverage of the teacher's sequence-level distribution; (ii) Misalignment between the teacher's output distribution and the student's learning capacity; and (iii) Exposure bias stemming from teacher forcing during training versus autoregressive inference at test time. In the following sections, we detail our rationale and empirical explorations for the three limitations.

## 3 Temperature-scheduled Learning

According to Equation 4, SFT on teacher-generated data can implicitly convey the teacher's distributional information. Consequently, the strategy for selecting samples—used to approximate the teacher's output distribution—plays a pivotal role in how effectively this knowledge is transferred to the student. However, most existing methods for long CoT distillation overlook this consideration, typically relying on random sampling (RS) from the teacher followed by quality-based filtering prior (Yan et al., 2025; Lei et al., 2025). This approach tends to produce samples that cover only a small subset of the teacher's modes, thereby under-utilizing the rich latent information embedded in the teacher's distribution. A natural remedy is to increase the sampling temperature, which flattens the teacher's distribution and leads to better coverage of its full mode structure (Holtzman et al., 2020; Jang et al., 2017).

As shown in Figure 3(a), we visualize the probability distribution of teacher-generated responses sampled at different temperatures. At lower temperatures (T=0.6), the resulting distribution becomes sharper and more peaked, concentrating most probability mass in a narrow range of high-likelihood responses. In contrast, higher-temperature (T=1.0) sampling yields a flatter and broader density, substantially increasing the covered probability range and markedly enhancing data diversity. However, we observe that high-temperature sampling introduces many rare teacher modes or potentially noisy samples. When the student model has limited capacity and exhibits a substantial architectural or behavioral gap from the teacher, it struggles to effectively learn from such heterogeneous data. Figure 3(b) compares the SFT training loss using datasets sampled at different temperatures. Specifically, we randomly sampled 50K math responses from the gpt-oss-120b teacher model at low and high temperatures, and fine-tuned the Qwen3-4B-Instruct-2507 student model on each dataset separately. The low-temperature dataset enables rapid convergence to a lower loss with a smooth downward trajectory, whereas the high-temperature dataset makes learning difficult: the loss stays higher.

Despite being more challenging to learn from, training on data sampled at temperature T=1.0 consistently outperforms that sampled at T=0.6, as evidenced in Table 1. Training on T=1.0 samples yields a +1.4 absolute improvement on AIME24 and an even larger +4.2 gain on the more representative and challenging AIME25. This indicates that—even under more difficult optimization dynamics and slower convergence—broader coverage of the teacher's output modes can lead to substantially greater gains for the student model. It also underscores the crucial impact of the sampling strategy in determining the efficacy of sequence-level distillation. To further evaluate the impact of high-temperature sampling,

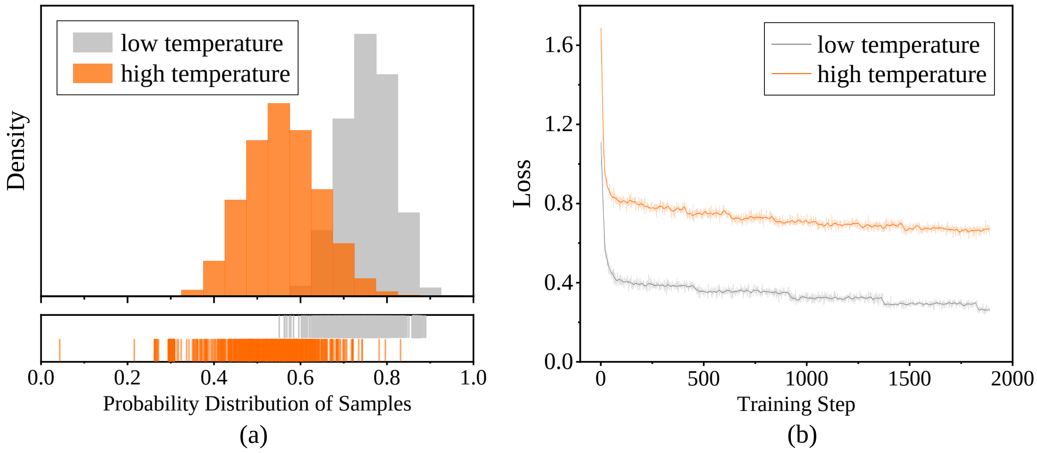

Figure 3: Comparison of probability distribution and training loss with data sampled from **gpt-oss-120b** under different temperatures. We randomly sampled 50K mathematical reasoning responses at both low (T=0.6) and high (T=1.0) temperatures. To characterize the overall likelihood of a response, we compute **the geometric mean of its token-level probabilities**. (a) Probability distributions of sampled responses: the upper panel displays the density of probability distribution, while the lower panel shows the probability intervals covered by the sampled responses. (b) SFT training loss curves for the student model trained on responses sampled at these temperatures.

we scaled the dataset size to 100K samples (doubling the 50K baseline). However, this increase in data volume yields only marginal improvements: as shown in Table 1, the 100K T=1.0 setup achieves no gain on AIME24 relative to 50K, and only a +2.8 improvement on AIME25. This suggests that the student's capacity to absorb diverse teacher behaviors becomes a bottleneck; adding more high-temperature samples does not translate into proportional performance gains.

Table 1: Performance comparison with different temperature settings.

| **Settings of training data** | **AIME24** | **AIME25** |
|---|---|---|
| **Teacher:** gpt-oss-120b **Student:** Qwen3-4B-Instruct-2507 | | |
| 50K Math + RS ($T = 0.6$) | 81.7 | 71.9 |
| 50K Math + RS ($T = 1.0$) | 83.1 | 76.1 |
| 100K Math + RS ($T = 1.0$) | 83.1 | 78.9 |
| 50K Math + RS ($T = 1.0$) w/ cold start ($T = 0.6$) | **85.2** | **81.3** |
| **Teacher:** Qwen3-Next-80B-A3B-Thinking **Student:** Qwen3-4B-Instruct-2507 | | |
| 25K Math + RS ($T = 0.6$) | 79.0 | 71.3 |
| 25K Math + RS ($T = 1.0$) | 82.9 | 70.2 |
| 25K Math + RS ($T = 1.0$) w/ cold start ($T = 0.6$) | **83.1** | **73.1** |

Based on these observations, we propose a temperature-scheduled learning pipeline for sequence-level distillation, an approach inspired by classic logit-based distillation (Caron et al., 2021; Zhou et al., 2021), which we extend to the SFT setting. We begin by sampling from the teacher at a low temperature, yielding a concentrated set of high-probability, easier-to-learn modes. We then switch to a higher temperature to collect more diverse samples that capture rarer teacher modes and richer latent information, albeit at the cost of increased learning difficulty. Accordingly, we use the low-temperature data to cold-start the student and then continue training with the high-temperature data. As an analogy, this can be intuitively viewed as an easy-to-hard curriculum temperature schedule (Li et al., 2023b), or equivalently, a "reverse" version of temperature annealing, a strategy that increases temperature over training, metaphorically inverting the cooling process of conventional annealing. As shown in Table 1, cold-starting with 50K samples at temperature 0.6 followed by continued training on another 50K samples at temperature 1.0 yields significant performance gains over all static-temperature baselines. This demonstrates that our strategy successfully reconciles two objectives: (i) facilitating stable early-stage learning, and (ii) broadening coverage of the teacher's output distribution, thereby transferring more valuable latent knowledge from the teacher to the student.

We further validate this approach across a broad range of domains and diverse teacher–student model

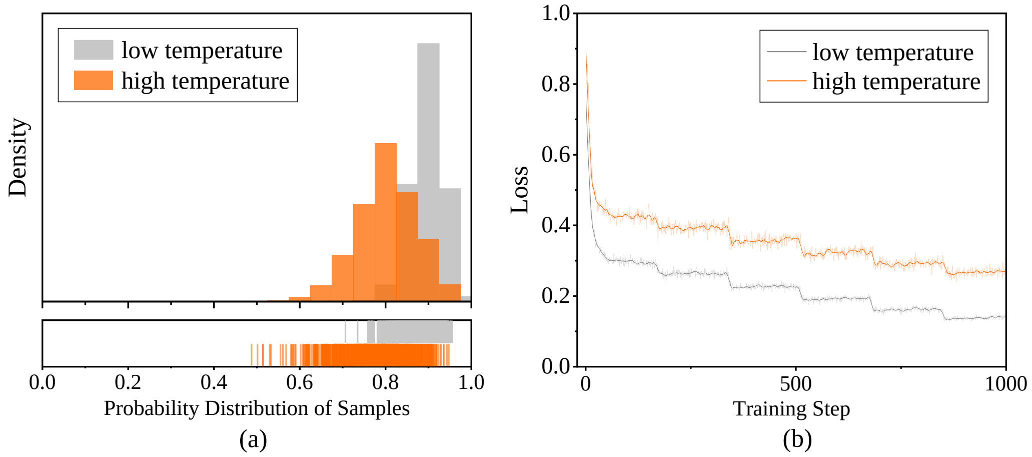

Figure 4: Comparison of probability distribution and training loss with data sampled from **Qwen3-Next-80B-A3B-Thinking** under different temperatures.

pairs. In the lower part of Table 1, we use Qwen3-Next-80B-A3B-Thinking as the teacher and randomly sample a small-scale of 25K math reasoning traces at temperatures 0.6 and 1.0, respectively. Continuing training—starting from a cold initialization with T=0.6 samples—and incorporating additional T=1.0 samples yields performance gains of +4.1 on AIME24 and +1.8 on AIME25, which also surpasses training on T=1.0 data alone. As shown in Figure 4, the shift in response probability distributions between T=0.6 and T=1.0 is markedly smaller for Qwen3-Next-80B-A3B-Thinking than for gpt-oss-120b. When training on data from both temperatures, the loss on T=1.0 samples remains higher, but the gap relative to T=0.6 is much narrower than for gpt-oss-120b. Nevertheless, temperature-scheduled learning still improves performance, indicating that samples drawn at different temperatures are complementary. In practice, optimal temperature combinations can be selected based on the model's evaluation performance and the response probability distributions observed across temperature settings. In Table 2, we further evaluate our approach under multi-domain mixed training. Using a small-scale mixture of math, code, and science reasoning data, we show that temperature-scheduled learning remains effective: it delivers substantial gains on AIME25, LiveCodeBench v6, and GPQA Diamond over training with data generated solely at T=0.6 or T=1.0. With the total training data held constant, it also outperforms the T=1.0-only baseline on AIME25 and GPQA Diamond, while achieving comparable performance on LiveCodeBench v6 (potentially attributable to the relatively small proportion of code data in the mixture, where further scaling could still yield noticeable improvements).

Table 2: Performance comparison with different settings of training data across domains.

| **Settings of training data** | **AIME25** | **LCB v6** | **GPQA-D** |
|---|---|---|---|
| **Teacher:** gpt-oss-120b **Student:** Qwen3-4B-Instruct-2507 | | | |
| 25K Math + 10K Code + 10K Science + RS ($T = 0.6$) | 74.6 | 44.1 | 65.5 |
| 25K Math + 10K Code + 10K Science + RS ($T = 1.0$) | 75.2 | 47.3 | 65.4 |
| 50K Math + 20K Code + 20K Science + RS ($T = 1.0$) | 75.8 | **51.3** | 65.4 |
| 25K Math + 10K Code + 10K Science + RS ($T = 1.0$) w/ cold start ($T = 0.6$) | **77.5** | 51.0 | **66.4** |

## 4 Divergence-aware Sampling

Despite employing temperature-scheduled learning to broaden coverage of the teacher's modes, the student still struggles to align with the teacher's sequence-level distribution. Classical logit distillation leverages teacher logit distribution to precisely calibrate the student's token-level probabilities, increasing or decreasing them as needed (Gu et al., 2024; Agarwal et al., 2024). By contrast, SFT on teacher-generated data typically amplifies the probabilities of all target tokens relative to the student's current predictions. This can induce misleading gradients: for tokens assigned low probabilities by the teacher but high probabilities by the student, SFT erroneously pushes the student's probabilities even higher, thereby driving them away from the teacher's distribution. This discrepancy motivates a core question: How can we identify a teacher-derived sequence-level distribution that is better aligned with the student model's learning capacity?

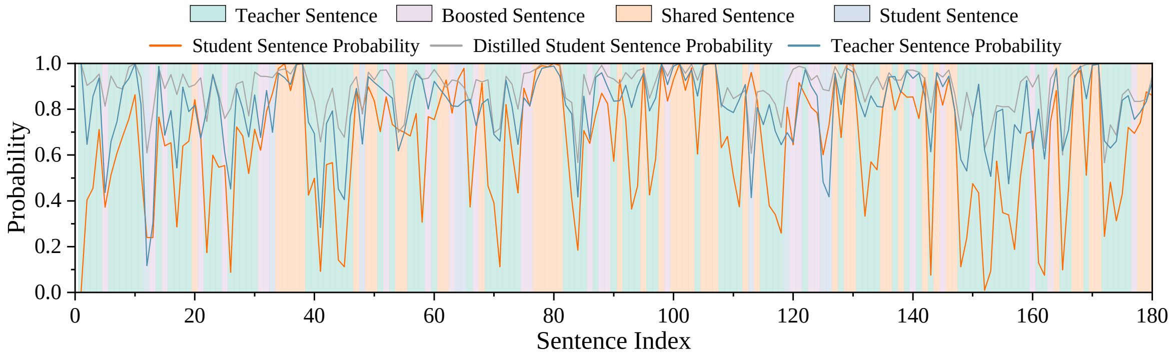

Figure 5: Joint comparison of the three models' predicted probabilities. An example of output probabilities: the x-axis indexes sentences, and the y-axis shows predicted probabilities. Foreground lines plot the probabilities of the three models, while background colors indicate the inferred source of each sentence. By comparing probability differences, every sentence is categorized into one of four source types.

To identify an effective sequence-level target distribution from the student's perspective, we introduce a distribution decomposition and analysis framework (Liu et al., 2025): each sequence-level response is decomposed into consecutive sentences and the corresponding sentence-level generation probabilities are computed for both the teacher and the student; by quantifying the probability discrepancy on each shared sentence, we categorize distinct behavioral patterns; finally, we systematically analyze these patterns (i.e., components) and establish their empirical relationship to effective student learning.

Concretely, following the experimental setup in Section 3, we first sample responses from the distilled model (i.e., the trained student model) on test-set prompts and segment each response into sentences. **This sentence-level analysis ensures the broad applicability of our method across heterogeneous model families**—unlike approaches such as on-policy distillation, which typically requires all models to share the same tokenizer and vocabulary (a constraint imposed by its reliance on token-level supervision). Then, we feed these samples to the teacher model, the pre-distillation student model (hereafter, the "student model"), and the post-distillation student model (hereafter, the "distilled model"). For each sentence of the response data, we compute its probability under each of the three models as the geometric mean of per-token probabilities in this sentence. As illustrated in Figure 5, we observe that the sequence-level distribution admits a natural decomposition into four well-defined distribution types (each corresponding to a distinct sentence category). Let $p_T$, $p_S$ and $p_D$ denote the predicted probabilities of the teacher, student, and distilled models for the same sentence, respectively. Based on the relative magnitude discrepancies of these probabilities, we define the following distribution (or sentence) types:

- Student-originated sentences (hereafter referred to as Student Sentence) and teacher-originated sentences (hereafter referred to as Teacher Sentence): When there is a large discrepancy between $p_S$ and $p_T$, the distilled model still outputs the sentence, suggesting the sentence is more consistent with the model assigning the higher likelihood. For example, if $p_T \gg p_S$ and distilled model nevertheless produces the sentence, it is more likely teacher-originated. **Moreover, when $p_T \gg p_S$, the student can relatively freely increase its probability under SFT without concern about misleading gradients. Intuitively, this type of pattern is more likely to apply under our current distillation setup.** Note that a Teacher Sentence does not imply that the action is entirely absent from the student model, but rather that it is primarily originated from the teacher. The same applies to a Student Sentence.
- Pre-existing sentences in both pre-distillation student model and teacher model, not enhanced by distillation (hereafter referred to as Shared Sentence): The output probabilities for these sentences are similar across all three models. This indicates that these sentences are already well-supported by both the pre-distillation student and the teacher, and that distillation does not materially change their probabilities or increase inter-model distribution discrepancies.
- Pre-existing sentences boosted through distillation (hereafter referred to as Boosted Sentence): Similar to the second type, $p_T$ and $p_S$ remain close, but $p_D$ differs significantly (and $p_D$ is typically higher in practice, since trajectories are sampled from the distilled model). These sentences also exist in both the teacher and the student before distillation, but their probabilities are amplified by training on distilled data.

Having decoupled the output distributions, we next investigate which distribution types are most conducive to the student model's learning (i.e., those that best support effective knowledge acquisition).

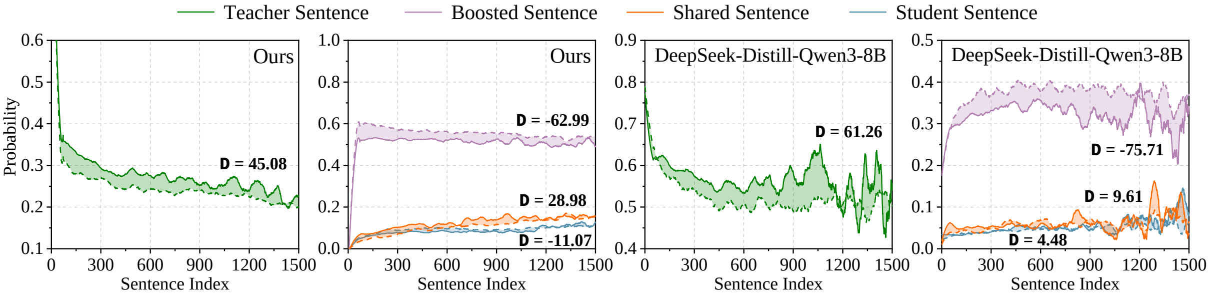

Figure 6: Position-wise distribution over the four sentence types for our internally trained model (left two panels) and the open-source DeepSeek-Distill-Qwen3-8B (right two panels). **The x-axis denotes the sentence position, and the y-axis denotes the predicted probabilities of the four sentence types. Solid lines (——) indicate probabilities when the answer is correct, while dashed lines(- - -) indicate probabilities when the answer is incorrect.** Δ denotes the area difference between the solid and dashed curves, which reflects the influence of each sentence type on answer correctness.

To this end, we assess effective learning by analyzing the correlation between the four distribution types and test-set answer correctness. Specifically, for each sentence position, we compute the probability that the distilled model assigns to each distribution type. Since solutions often contain multiple sentences (correct answers typically contain fewer sentences than incorrect ones), analyzing at the sentence-position-level, rather than at the full solution level, allows us to focus more directly on the distribution types themselves and mitigate confounding effects arising from sentence position. For example, to estimate the probability of the Teacher Sentence at the third sentence position, we calculate the fraction of third sentences that are categorized as Teacher Sentence, across all correct and incorrect model outputs. Notably, the number of sentences per answer varies, limiting data availability at later positions. To ensure statistical reliability, we therefore focus primarily on earlier sentence positions, where sufficient samples exist. We also replicate this analysis on the open-source model DeepSeek-Distill-Qwen3-8B (Guo et al., 2025) to ensure generalizability. **As shown in Figure 6, across models, Teacher Sentences tend to receive higher probabilities in correct answers**, evidenced by the light-green solid line (——) persistently lying above the light-green dashed line (- - -). This is as expected: since the teacher model performs better on the test set, aligning the student's outputs with teacher-preferred responses enhances learning efficacy and, consequently, the likelihood of generating correct answers. In contrast, we find that Shared Sentence and Student Sentence occur with low probability and exert a relatively minor influence. For Boosted Sentence, we observe a potential negative correlation between Boosted Sentences and test-set accuracy. We conjecture that this possibly stem from suboptimal misleading gradients. More importantly, the distillation pipeline only admits the teacher and student models prior to training, rendering it impossible to directly identify Boosted Sentences. We therefore focus primarily on Teacher Sentences in the remainder of this work.

Building on the above analysis, a natural idea is to emphasize, during training, patterns that are more indicative of answer correctness. Although the full distribution-decomposition framework requires output probabilities from three models (the teacher, the student, and the distilled model) to identify the most effective distribution post hoc, we show that Teacher Sentences/Student Sentences can be identified prior to training: Teacher Sentences/Student Sentences are those sentences for which the teacher assigns significantly higher/lower output probabilities than the student. Therefore, we propose divergence-aware sampling (DAS), which prioritizes training examples rich in Teacher Sentences and thereby implicitly targets a teacher-derived sequence-level distribution better aligned with the student's learning capacity (Liu et al., 2025). This sampling distribution naturally mitigates misleading gradients and facilitates more effective knowledge transfer from teacher to student. **Notably, our method only requires, for each token in the teacher-generated response, its predicted probability by both the teacher and the student**. The teacher-side probabilities are naturally obtained during sampling—and are often exposed even by many closed-source APIs—while the student-side probabilities are readily computed from the local model. In contrast, classical logit-based distillation necessitates the teacher's full-vocabulary logits (i.e., probabilities over the entire vocabulary) at every position. **Even recent on-policy distillation methods—when simplified to operate on token-level probabilities—still require, for every token in the student's generated outputs, the corresponding probabilities under both models. Critically, the teacher-side probabilities for the student's outputs are typically unavailable for proprietary models**.

Building on the experimental setup in Section 3, we conduct a controlled comparison between DAS and random sampling (DS) under an identical sampling budget. As shown in Table 3, DAS consistently achieves higher test performance, and in several cases, even surpasses the results obtained by RS after scaling up its data volume. This demonstrates that DAS effectively identifies teacher-generated sequences

Table 3: Performance comparison with different settings of training data (RS vs. DAS).

| Settings of training data | AIME24 | AIME25 |
|---|---|---|
| **Teacher:** gpt-oss-120b **Student:** Qwen3-4B-Instruct-2507 | | |
| 50K Math + RS ($T = 0.6$) | 81.7 | 71.9 |
| 50K Math + DAS ($T = 0.6$) | **83.3** | **74.2** |
| 50K Math + RS ($T = 1.0$) | 83.1 | 76.1 |
| 100K Math + RS ($T = 1.0$) | 83.1 | 78.9 |
| 50K Math + DAS ($T = 1.0$) | **85.0** | **79.2** |
| **Teacher:** Qwen3-Next-80B-A3B-Thinking **Student:** Qwen3-4B-Instruct-2507 | | |
| 25K Math + RS | 79.0 | 71.3 |
| 25K Math + DAS | **82.5** | **71.9** |

Table 4: Performance comparison with different settings of training data (RS vs. DAS across domains).

| Settings of training data | AIME25 | LCB v6 | GPQA-D |
|---|---|---|---|
| **Teacher:** gpt-oss-120b **Student:** Qwen3-4B-Instruct-2507 | | | |
| 25K Math + 10K Code + 10K Science + RS | 74.6 | 44.1 | 65.5 |
| 25K Math + 10K Code + 10K Science + DAS | **75.6** | **47.3** | **65.7** |

whose distribution is better aligned with the student's learning capacity. Further, as shown in the lower part of Table 3 and in Table 4, DAS maintains a clear advantage over random sampling across different teacher models and domains, validating the generalizability of the DAS method.

Finally, DAS does not require re-sampling data for every new student model. For instance, as demonstrated in Section 7, data curated to match the learning capacity of the Qwen3-4B-Instruct-2507 student model generalizes effectively to the Qwen3-30B-A3B-Instruct-2507 student model.

## 5 Mixed-policy Distillation

In the previous stages, we employed off-policy methods to approximate the teacher's sequence-level distribution through high-quality data generation. Nevertheless, we find that the resulting student model still suffers from exposure bias (Ranzato et al., 2016): during training, the student is conditioned on the teacher's prefix using teacher forcing, whereas at inference time, it must rely on its own autoregressive predictions, leading to a distribution mismatch.

To empirically investigate this phenomenon, we use the student model trained in the previous round (50K DAS sampling, T=0.6; see Table 3) to re-generate the training data within its own context, in order to examine whether the student model is overly reliant on the teacher's context. During inference, we set the maximum generation token length to 1.5 times the length of the teacher-provided solution, in order to compare the differences between the teacher's reference response and the student's self-generated counterpart. Figure 7 plots the cut-off rate of the student's generated responses across different training-response lengths, where a higher cut-off rate indicates greater divergence between student and teacher behavior. The results reveal that, even on the training data, the student still exhibits substantial deviations from the teacher, and this discrepancy becomes increasingly pronounced as the length of the training response grows. This observation confirms that training with teacher forcing under longer teacher prefixes exacerbates exposure bias.

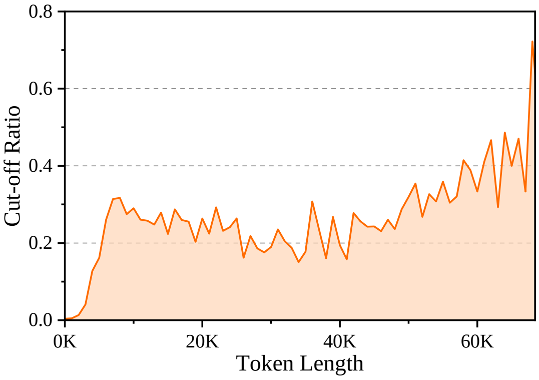

Figure 7: The ratio between cut-off responses under different token lengths.

To overcome these limitations, Chen et al. (2025a) present that on-policy data collection is an effective alternative method. Accordingly, we propose a mixed-policy distillation approach that synergistically combines off-policy and on-policy signals. Specifically, we first use the student model from the previous training round to re-generate responses for the training queries, and then identify instances that differ substantially from the teacher's outputs, e.g., solutions that have been cut off in Figure 7. For these data points, we randomly cut off the solutions generated by the student and prompt the teacher to continue the generation, thereby enabling the teacher to provide targeted guidance on the student's errors.

We present an ablation study of the proposed mixed-policy distillation in Table 5. As described above, we collect 7.7K mixed-policy samples, and train the model with only one epoch across these samples. Our baseline is the model trained on the 50K DAS-generated dataset at temperature T=0.6 (Table 3). In addition, we investigate a masking variant, where student-generated portions are masked out, and only teacher-completed segments are retained for training.

To maintain a balanced proportion between mixed-policy and off-policy data during training, we introduce 20K additional off-policy samples for joint training with the mixed-policy data. Our main experimental observations are as follows: (1) The results indicate that our mixed-policy dataset, despite containing only 7.7K samples, is capable of enhancing the model's performance. (2) The masking variant tends to yield worse performance. This is because masking removes the on-policy segments generated by the student, leaving only the off-policy segments from the teacher for training. The observed performance drop in this setting further demonstrates the importance of incorporating on-policy data during distillation. As shown in Table 7, we also validate the effectiveness of the mixed-policy distillation approach in our final training pipeline. Incorporating even a small amount of mixed-policy data yields measurable gains across strong models and diverse domains. These results motivate continued exploration of this promising direction in the future.

Table 5: Ablation of the mixed-policy distillation method. #Num: number of mixed-policy data.

| #Num | Mask | AIME24 | AIME25 |
|---|---|---|---|
| *Baseline* | | | |
| 50K DAS ($T = 0.6$) | — | 83.3 | 74.2 |
| *Mixed-Policy Variants* | | | |
| 7.7K | ✓ | 80.8 | 72.3 |
| 7.7K | ✗ | **83.3** | **74.8** |

## 6 Overall Training Recipe

In this section, we detail the concrete implementation of DASD-4B-Thinking. Our pipeline comprises (i) question collection, (ii) candidate response sampling, (iii) filtering to remove low-quality responses, and (iv) multi-stage training on the curated dataset. Beyond our core innovations, we also integrate well-established practices from prior work in this line to ensure the overall quality of responses. We present each component in turn and describe the associated design choices in detail below.

### 6.1 Question Collection

Our goal is to collect challenging questions spanning a diverse set of domains in order to obtain responses that demonstrate meaningful reasoning. To this end, we select four representative domains: mathematical reasoning, code generation, scientific reasoning, and instruction following. To gather training questions, we utilize a variety of publicly available open-source datasets (Chen et al., 2025b; Nvidia, 2024; Ji et al., 2025), which summarize numerous high-quality questions.

- **Mathematical Reasoning.** Our math questions are primarily sourced from the supervised fine-tuning dataset of NVIDIA AceReason (Chen et al., 2025b). We experimented with different question scales during training and, considering the diminishing returns with larger question scales, ultimately sampled about 105K questions. These questions include those from original sources such as NuminaMath-CoT (Li et al., 2024) and the Art of Problem Solving (AoPS) community forums.
- **Code Generation.** Our code questions primarily come from the OpenCodeReasoning dataset (Ahmad et al., 2025), with data sources including TACO (Li et al., 2023a), CodeContests (Li et al., 2022), APPs (Hendrycks et al., 2021), and Codeforces.
- **Scientific Reasoning.** Scientific questions are primarily collected from NVIDIA's OpenScience Reasoning dataset (Nvidia, 2024), which consists entirely of multiple-choice question-answer pairs. We prioritize questions with longer example answers, as they tend to better elicit the model's reasoning abilities.
- **Instruction Following.** Instruction-following questions are sourced from AM-DeepSeek-R1-Distilled-1.4M (Ji et al., 2025), which contain a large number of subjective instructions from various datasets.

### 6.2 Response Sampling

We select a representative student-teacher pair, Qwen3-4B-Instruct-2507 and gpt-oss-120b, as our primary model pair, highlighting the broad compatibility of our approach across diverse model families and architectures.

During sampling, we leverage the teacher's high-level reasoning capabilities to generate multiple candidate samples for each question. To enhance coverage of the teacher model's behavior, we follow the methodology outlined in Section 3, sampling multiple responses per question at both low and high temperatures. Furthermore, we apply divergence-aware sampling as described in Section 4 to prioritize examples that better support student learning while preserving the correct gradient directions.

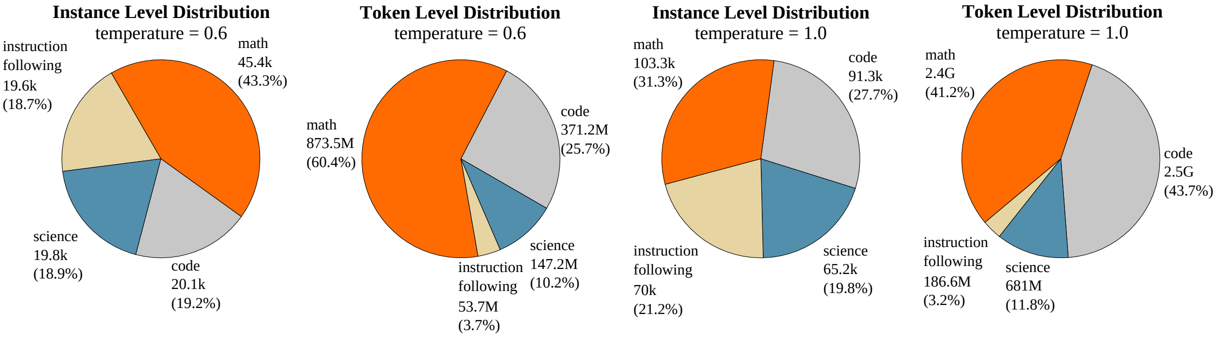

Figure 8: Data distribution.

### 6.3 Response Filtering

Beyond our core innovations, we employ stringent response filtering strategies to remove low-quality samples:

- **Length-based Filtering.** We compute response lengths using the student model's tokenizer, whose segmentation may differ substantially from the teacher's. Responses that exceed the training context length are discarded.
- **Structure-based Filtering.** gpt-oss-120b occasionally invokes built-in tools to answer questions. Since our current focus is on distilling long CoT reasoning capabilities—and function calling is deferred to future work—we explicitly detect and filter out any responses containing function calls. Additionally, we require every response to include both a thinking process and a final answer. gpt-oss-120b uses a harmony template to separate the reasoning trace from the final answer; for student-model compatibility, we post-process the original outputs by replacing the native delimiters with “$< think >$” and “$< /think >$”.
- **Repetitive Content Filtering.** We observe that gpt-oss-120b tends to generate repetitive content, particularly at lower temperatures, including repeated paragraphs, sentences, or phrases within a single response. Such repetitions can induce the trained student to produce endlessly repetitive and excessively verbose outputs during inference. To mitigate this, we rigorously filter out data containing repetitive content using regular expressions and n-gram matching, preventing the student from internalizing undesirable patterns.

Finally, we acquire a total of 105K low-temperature (T=0.6) and 330K high-temperature (T=1.0) responses. The resulting data distribution across various domains is shown in Figure 8.

### 6.4 Multi-stage Training

As illustrated in Figure 2, our training pipeline comprises two main stages: temperature-scheduled learning and mixed-policy distillation. During temperature-scheduled learning, the sampling data used to train DASD-4B-Thinking undergoes a two-stage filtering and training process to better capture the teacher model's distribution while effectively supporting the student model's learning. In the subsequent mixed-policy distillation, we construct mixed-policy data via on-policy rejection sampling and off-policy teacher revision. This hybrid strategy mitigates exposure bias by providing targeted, error-aware supervision.

#### 6.4.1 Temperature-scheduled Learning

To better represent the teacher model's sequence-level distribution, we follow Section 3, and perform two-stage SFT on Qwen3-4B-Instruct-2507: first using low-temperature sampling data, followed by high-temperature sampling data. Training configurations are kept identical across both stages. We use an initial learning rate of 5e-5 that decays to 1e-5 via a cosine scheduler. We set the cutoff length to 64K and employ greedy sequence packing to accelerate training. Given the substantial GPU memory demands of 64K-context training, we leverage ZeRO-3 optimization together with Liger kernels to reduce memory consumption. Training is conducted with a global batch size of 64 over 6 epochs, and we observe consistent performance improvements across epochs.

#### 6.4.2 Mixed-policy Distillation

To mitigate the exposure bias, and inspired by the effectiveness of on-policy data (Chen et al., 2025a), we propose a mixed-policy revision protocol. Starting from the DAS-curated training set, we sample 50K questions. Each question is fed to the student model trained in the previous stage to generate responses. To align with the teacher's reference length, we cap the student's generation at 1.5 times the token count of the corresponding teacher response to the same question. Among the above student-generated solutions, we identify 15K truncated responses. For each truncated response, we discard the portion after a randomly selected position located beyond half of its total length, and then employ the teacher model to rewrite this discarded part. The teacher continuations that pass predefined quality filters are retained for the student's further fine-tuning, yielding a total of mixed-policy 12.7K data samples.

## 7 Experimental Evaluation

### 7.1 Benchmarks

We evaluate models on five complementary reasoning benchmarks:

- **AIME24&AIME25** (AIME, 2025): Problem sets from that year's American Invitational Mathematics Examination (AIME) I/II, each comprising 30 challenging problems, focusing on mathematical reasoning and requiring the correct final answer.
- **GPQA Diamond (GPQA-D)** (Rein et al., 2023): A graduate-level benchmark of 198 expert-written multiple-choice questions spanning physics, chemistry, and biology, emphasizing "Google-proof" deep academic reasoning.
- **LiveCodeBench (LCB)** (Jain et al., 2024): A continuously updated coding benchmark that mitigates data contamination through strict temporal partitioning. Beyond code generation, it evaluates self-repair, executable correctness, and test-output prediction. The benchmark is released in time-based snapshots; **v5** contains problems collected from October 2024 to February 2025, and **v6** covers problems collected from February 2025 to May 2025.

### 7.2 Baselines

We release our model, , and evaluate its performance on five public reasoning benchmarks: AIME24, AIME25, LCB and GPQA-D. We compare against two families of state-of-the-art open-source models serving as our primary baselines. All baselines were selected for their demonstrated strength in reasoning and complex problem solving, ensuring a relevant and competitive evaluation context for . Particularly, we categorize these baselines based on the accessibility of their training data. This enables a dual-perspective assessment of : (i) against top-performing models trained on private or proprietary data, and (ii) against leading models representing fully transparent, reproducible research.

- **Open-Weights Only.** This group comprises models that release their weights publicly but keep their training data as proprietary. These models often represent the best performance available from models where the full training process is not disclosed. Our comparison set includes:
  - The **Qwen3 series** (4B-Thinking-2507, 8B, 14B, 32B) (Yang et al., 2025): The world's most popular model family featuring mainline models (8B, 14B, 32B) with switchable reasoning modes, alongside a dedicated 4B-Thinking variant optimized for complex reasoning over long contexts.
  - **DeepSeek-R1-0528-Qwen3-8B** (Guo et al., 2025): A specialized 8B reasoning model distilled from the powerful teacher model DeepSeek-R1-0528 into a Qwen3-8B backbone.
  - The **GLM-Z1 series** (32B-0414, 9B-0414) (Zeng et al., 2024): Models built on the GLM-4 architecture, enhanced via extended reinforcement learning for mathematical, logical, and coding reasoning.

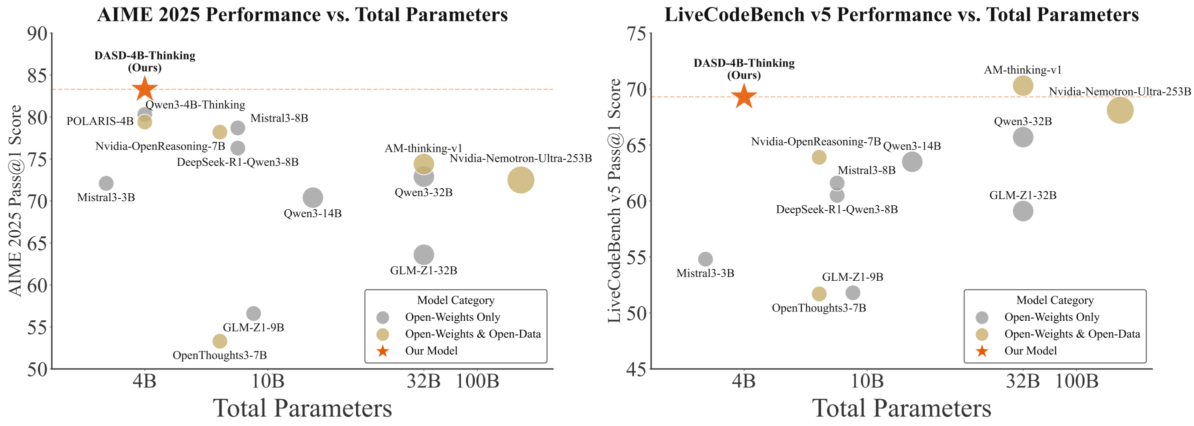

Figure 9: Performance versus model size on AIME25 (left) and LCB v5 (right). Each point denotes a model, with the x-axis representing model size (in number of parameters) and the y-axis indicating benchmark score. Points positioned toward the top-left indicate superior efficiency—i.e., higher performance at smaller scale.

  - The **Mistral 3 series** (3B, 8B) (Mistral AI Team, 2025): Compact open-weight models that emphasize efficient reasoning, strong math and coding capabilities, and multilingual generalization, suitable for low-latency, low-memory deployments.

- **Open-Weights & Open-Data.** This group releases both model weights and the curated reasoning datasets, enabling full reproducibility and fostering community-wide study of reasoning acquisition. Included models are:
  - **AM-thinking-v1** (Ji et al., 2025) and **OpenThoughts3-7B** (Guha et al., 2025): These models demonstrate different open-data strategies. AM-thinking combines SFT on **2.9M** distilled examples with RL on a Qwen2.5-32B base; OpenThoughts3 achieves strong 7B-level performance via SFT alone on its publicly released **1.2M** high-quality reasoning traces.
  - The **Pai-DistillQwen-ThoughtY series** (Cai et al., 2025): A set of compact models (4B, 8B) distilled from DeepSeek-R1-0528 using **365K** curated examples, accompanied by full dataset release.
  - **POLARIS-4B-Preview** (An et al., 2025): A model based on Qwen3-4B that highlights the effectiveness of scaling up RL on public data to significantly improve complex, long-context reasoning.
  - The **Nemotron family** (Bercovich et al., 2025): This collection includes OpenReasoning-Nemotron-7B, distilled from DeepSeek-R1-0528 on a massive **30M** example open dataset, and Nemotron-Ultra-253B, a large-scale model targeting high-end reasoning tasks.

### 7.3 Evaluation Setup

All evaluations were conducted under a unified setup. We consistently set the temperature to 1.0 and top-$p$ to 1.0. For every benchmark, we sampled 64 responses per question and reported the average accuracy to ensure reliable and stable evaluation results. Given the extreme difficulty of AIME24 and AIME25, we set the maximum generation length to 102,400 tokens; For LiveCodeBench and GPQA-D, the limit was set to 81,920 tokens.

### 7.4 Main Results

Table 6 summarizes our evaluation, robustly validating the effectiveness of our proposed framework. The results demonstrate that, with our enhanced sequence-level distillation pipeline, complex reasoning capabilities can be efficiently transferred from a large teacher to a lightweight 4B-parameter student, yielding state-of-the-art performance for its scale. Notably, as shown in Figure 9, not only outperforms all comparable-size models but also surpasses significantly larger counterparts (e.g., 32B) on multiple key benchmarks, highlighting both the effectiveness of our approach and the high efficiency of our training data.

**Mathematical Reasoning (AIME24, AIME25).** On the most challenging mathematical reasoning benchmarks, achieves remarkable scores of **83.3** on AIME25 and **88.5** on AIME24, establishing state-of-the-art

Table 6: **Comparison across AIME24, AIME25, LiveCodeBench (v5/v6), and GPQA-D.**

| | **AIME24** | **AIME25** | **LCB v5** | **LCB v6** | **GPQA-D** |
|---|---|---|---|---|---|
| | | *Open-Weights Only* | | | |
| Qwen3-4B-Thinking-2507 | - | 81.3 | - | 55.2 | 65.8 |
| Qwen3-14B | 79.3 | 70.4 | 63.5 | - | 64.0 |
| Qwen3-32B | 81.4 | 72.9 | 65.7 | - | 68.4 |
| DeepSeek-R1-0528-Qwen3-8B | 86.0 | 76.3 | 60.5 | - | 61.1 |
| GLM-Z1-32B-0414 | 80.8 | 63.6 | 59.1 | - | 66.1 |
| GLM-Z1-9B-0414 | 76.4 | 56.6 | 51.8 | - | 58.5 |
| Mistral3-3B | - | 72.1 | 54.8 | - | 53.4 |
| Mistral3-8B | - | 78.7 | 61.6 | - | 66.8 |
| | | *Open-Weights & Open-Data* | | | |
| AM-thinking-v1 | 85.3 | 74.4 | 70.3 | - | - |
| POLARIS-4B-Preview | 81.2 | 79.4 | - | - | - |
| OpenThoughts3-7B | 69.0 | 53.3 | 51.7 | - | 53.7 |
| Pai-DistillQwen-ThoughtY-4B | 76.7 | - | - | - | 56.1 |
| Pai-DistillQwen-ThoughtY-8B | 76.7 | - | - | - | 62.1 |
| NVIDIA-OpenReasoning-Nemotron-7B | 84.7 | 78.2 | 63.9 | - | 61.4 |
| NVIDIA-Nemotron-Ultra-253B | 80.8 | 72.5 | 68.1 | - | 76.0 |
| DASD-4B-Thinking (Ours) | **88.5** | **83.3** | **69.3** | **67.5** | **68.4** |

performance among all listed models. These scores confirm that  delivers top-tier reasoning capability, even when compared to larger scale models. This exceptional efficiency is visualized in Figure 9, where is clearly positioned in the top-left, achieving superior performance at a fraction of the parameter cost of its competitors.

Notably, ’s performance:

- In the “Open-Weights & Open-Data” category, our 4B model demonstrates clear superiority. It substantially outperforms the 32B AM-thinking-v1 on AIME25 (83.3 vs 74.4) and AIME24 (88.5 vs 85.3). This is particularly noteworthy given that AM-thinking-v1 was trained on 2.9M examples, whereas  achieves this result using only **448K** examples—a dataset roughly 6 times smaller. It also surpasses other strong open-data models, including NVIDIA-OpenReasoning-Nemotron-7B (84.7/78.2), which used a massive 30M dataset, and even NVIDIA-Nemotron-Ultra-253B (80.8/72.5), a model over 60 times its size.
- In the “Open-Weights Only” category,  also sets a new benchmark for compact reasoning models, decisively outperforming the strong Qwen3-4B-Thinking-2507 on AIME25 (83.3 vs. 81.3). It further exceeds several “Open-Weights Only” models of medium to large scale, including Qwen3-32B (81.4/72.9) and GLM-Z1-32B (80.8/63.6).

These results strongly demonstrate that our refined, data-efficient sequence-level distillation pipeline effectively enhances ’s reasoning capabilities, enabling it to match or surpass models 8 to 60 times larger in parameter count.

**Coding (LiveCodeBench).** In code generation,  scores **69.3** on LCB v5 and **67.5** on LCB v6, again demonstrating exceptional efficiency. On LCB v5, this score not only surpasses DeepSeek-R1-0528-Qwen3-8B (60.5) and Qwen3-14B (63.5), but also the NVIDIA-OpenReasoning-Nemotron-7B (63.9). On LCB v6, it outperforms the strong Qwen3-4B-Thinking-2507 (67.5 vs. 55.2). Our 4B model even surpasses Qwen3-32B (65.7), highlighting the high efficiency of our method in transferring complex code generation capabilities.

**Scientific QA (GPQA-D).**  achieves a remarkable **68.4** on GPQA-D. It not only outperforms its same-size counterparts but also closely approaches the performance of substantially larger models, e.g., Qwen3-32B (68.4) and NVIDIA-Nemotron-Ultra-253B (76.0). While GPQA is notoriously challenging for compact models due to its heavy reliance on parametric knowledge, our 4B model successfully narrows the gap to these massive baselines. This result strongly suggests that our innovations effectively maximize the utilization of limited capacity for scientific reasoning.

**Summary.** Overall,  delivers state-of-the-art performance across mathematical, coding, and scientific reasoning benchmarks, robustly validating the effectiveness and data efficiency of our sequence-level distillation framework. It provides new insights and a fresh perspective for developing compact, high-performing, and fully open reasoning models.

### 7.5 Ablations over training stages

Sections 3, 4, and 5 have validated the individual contributions of our three core components through isolated experiments. In this section, we complement those findings with a holistic ablation of the full training pipeline, examining how performance evolves after each sequential stage. Results are summarized in Table 7.

Table 7: **Ablations over training stages on AIME24, AIME25, LiveCodeBench v5, LiveCodeBench v6, and GPQA-D.**

| | AIME24 | AIME25 | LCB v5 | LCB v6 | GPQA-D |
|---|---|---|---|---|---|
| Qwen3-4B-Instruct-2507 | - | 47.4 | - | 35.1 | 62.5 |
| + Low-Temperature Training | 84.2 | 74.0 | 56.6 | 50.6 | 67.7 |
| + High-Temperature Training | 87.7 | 83.0 | 68.4 | 67.2 | 67.6 |
| + Mixed-Policy Distillation | 88.5 | 83.3 | 69.3 | 67.5 | 68.4 |

Starting from the Qwen3-4B-Instruct-2507 baseline, we observe consistent performance improvements across the three stages:

**Low-temperature training (with DAS)** delivers substantial initial gain, boosting AIME25 from 47.4% to 74.0% (+26.6%) and LCB v6 from 35.1% to 50.6% (+15.5%). This confirms that stable, low-variance gradient signals during early training are critical for establishing a solid reasoning foundation.

**High-temperature training (with DAS)** further enhances performance across key benchmarks, advancing LCB v5 by +11.8% and LCB v6 by +16.6%, while also providing a notable +9.0% gain on AIME25. This demonstrates that diverse exploration under higher temperature effectively expands the policy's solution coverage once a stable baseline has been established.

**Mixed-policy distillation** consistently yields performance gains even on top of an already strong model across all benchmarks (e.g., +0.8% on AIME24, +0.3% on AIME25, +0.9% on LCB v5, +0.3% on LCB v6, +0.8% on GPQA-D), supporting the effectiveness of mixed-policy distillation in addressing the exposure-bias issue with minimal training overhead.

### 7.6 Effect of Divergence-aware Sampling on Data Distribution

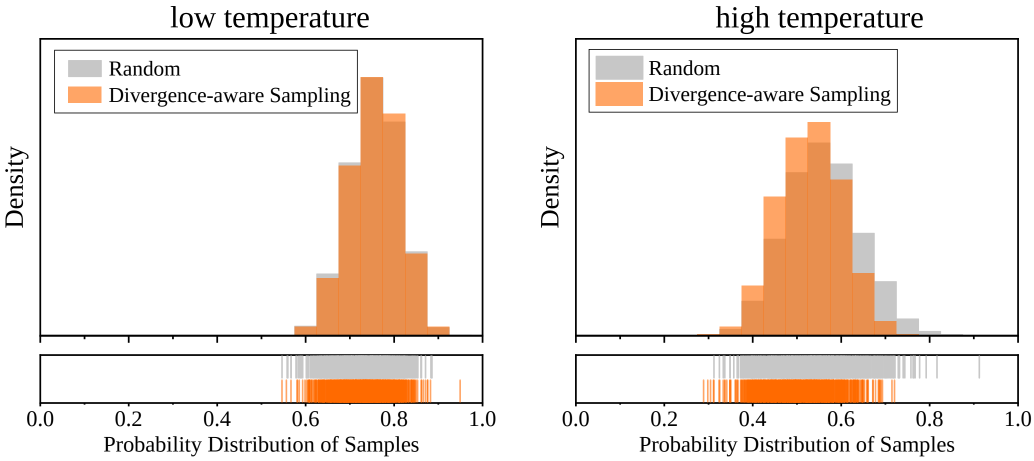

Figure 10: Comparison of response probability distributions with/without divergence-aware sampling using different temperatures.

To broaden coverage of the teacher's output modes, we employ temperature-scheduled learning to collect both low-temperature samples (sharper, more concentrated distributions that cover a narrower probability range around high-probability regions) and high-temperature samples (flatter, broader densities that capture rarer teacher modes). To better identify the target sequence-level distribution that supports effective student learning, we further apply divergence-aware sampling to both data subsets. As illustrated in Figure 10, divergence-aware sampling induces negligible perturbation to the underlying response probability distribution, confirming its orthogonality to temperature scheduling. This decoupling underpins the strong synergy observed when combining the two strategies, as evidenced by the substantial performance gains in Section 7.5.

### 7.7 Evaluation on MoE Models

To assess the scalability and architectural robustness of our distillation framework, we further extend it to Mixture-of-Experts (MoE) students, which increase model capacity while maintaining inference efficiency via sparse expert routing. We select Qwen3-30B-A3B-Instruct-2507 as the student model. For this investigation, we conduct a preliminary evaluation by applying **only the first stage** of our pipeline (i.e., Low-Temperature Training with DAS), yielding the model denoted as DASD-30B-A3B-Thinking-Preview. Crucially, to test the cross-architecture transferability of our data, we do not re-collect or re-curate training samples; instead, we directly reuse the exact dataset curated for the Qwen3-4B student. Table 8 summarizes the results. All baseline results are quoted from the corresponding official technical reports.

Table 8: **Performance comparison on MoE models.**

| **Model** | **AIME25** | **LCB v6** | **GPQA-D** | **Average** |
|---|---|---|---|---|
| gpt-oss-20b | 91.7 | 61.0* | 71.5 | 74.7 |
| Qwen3-30B-A3B-Thinking-2507 | 85.0 | 66.0 | 73.4 | 74.8 |
| NVIDIA-Nemotron-3-Nano-30B-A3B | 89.1 | 68.3 | 73.0 | 76.8 |
| DASD-30B-A3B-Thinking-Preview (Ours) | **86.7** | **72.8** | **72.3** | **77.3** |

* indicates the number is taken from the technical report of NVIDIA-Nemotron-3.

Despite being a preview version that has not been thoroughly trained, DASD-30B-A3B-Thinking-Preview already demonstrates strong competitiveness against powerful open MoE baselines. Compared with Qwen3-30B-A3B-Thinking-2507, it achieves robust gains across key benchmarks, boosting AIME25 to 86.7% (+1.7%) and LCB v6 to 72.8% (+6.8%), while maintaining competitive performance on GPQA-D. When evaluated against gpt-oss-20b, our model secures a substantial lead in coding tasks, elevating the LCB v6 score to 72.8% (+11.8%) and driving the overall average to 77.3% (+2.6%). We further compare with NVIDIA-Nemotron-3-Nano-30B-A3B (NVIDIA, 2025): according to its technical report, it is trained with a large-scale SFT corpus (18M) and additional RL, whereas our model uses only the 105K distilled dataset from the first stage of our pipeline, with no extra data or RL involved. Even under this much lighter training recipe, DASD-30B-A3B-Thinking-Preview delivers stronger coding performance on LCB v6 (+4.5; 72.8 vs. 68.3) and a higher average score (+0.5; 77.3 vs. 76.8), highlighting that careful pipeline design can achieve superior efficiency-quality trade-offs at MoE scale.

## 8 Conclusion and Future Work

In this technical report, we introduce DASD-4B-Thinking, a high-performance large reasoning model developed through large-scale knowledge distillation. We critically re-examine the prevailing paradigm of SFT on teacher-generated responses and identify three key limitations: (i) inadequate coverage of the teacher's sequence-level distribution; (ii) misalignment between the teacher's output distribution and the student's learning capacity; and (iii) exposure bias stemming from teacher forcing during training versus autoregressive inference at test time. To address these challenges, we propose a comprehensive data construction and training pipeline built upon three core innovations: temperature-scheduled learning, divergence-aware sampling, and mixed-policy distillation. Leveraging this pipeline and training on only 448K samples, DASD-4B-Thinking achieves state-of-the-art performance across the majority of reasoning benchmarks—consistently outperforming models of comparable scale and, notably, surpassing several larger counterparts. Complementing this, we also release DASD-30B-A3B-Thinking-Preview, a MoE variant that attains competitive or superior results across the same benchmarks. To promote reproducibility and community advancement, we open-source both the curated dataset and the trained models.

Looking ahead, we outline the following directions for future work. First, we will explore distribution-aware reweighting during SFT, leveraging the teacher model's sequence-level output probabilities to more faithfully approximate its target distribution, thereby improving both distillation effectiveness and data efficiency. Second, we aim to further refine the mixed-policy distillation approach to enhance training efficiency and stability. Finally, we plan to integrate complementary agentic capabilities—such as knowledge retrieval and tool use—to progressively develop more powerful, domain-adapted reasoning models capable of handling complex, real-world tasks.

## 9 Acknowledgement

We would like to thank the following individuals (listed in alphabetical order by last name) for their valuable discussions and for their help with experiments and evaluation:

Xin Guo, Chenxi Huang, Zhen Huang, Yao Liu, Rui Miao, Ge Teng, Jingyi Shen, Xu Shen, Chaoqun Wan, Yue Xin, Xiaosong Yuan.

## References

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

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *CoRR*, abs/2508.10925, 2025.

Wasi Uddin Ahmad, Sean Narenthiran, Somshubra Majumdar, Aleksander Ficek, Siddhartha Jain, Jocelyn Huang, Vahid Noroozi, and Boris Ginsburg. Opencodereasoning: Advancing data distillation for competitive coding. *CoRR*, abs/2504.01943, 2025.

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

Chenxin An, Zhihui Xie, Xiaonan Li, Lei Li, Jun Zhang, Shansan Gong, Ming Zhong, Jingjing Xu, Xipeng Qiu, Mingxuan Wang, and Lingpeng Kong. Polaris: A post-training recipe for scaling reinforcement learning on advanced reasoning models, 2025. URL https://hkunlp.github.io/blog/2025/Polaris.

Akhiad Bercovich, Itay Levy, Izik Golan, Mohammad Dabbah, Ran El-Yaniv, Omri Puny, Ido Galil, Zach Moshe, Tomer Ronen, Najeeb Nabwani, et al. Llama-nemotron: Efficient reasoning models. *arXiv preprint arXiv:2505.00949*, 2025.

Wenrui Cai, Chengyu Wang, Junbing Yan, Jun Huang, and Xiangzhong Fang. Reasoning with omnithought: A large cot dataset with verbosity and cognitive difficulty annotations. *CoRR*, abs/2505.10937, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE/CVF International Conference on Computer Vision*, pp. 9630–9640, 2021.

Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. *CoRR*, abs/2510.18874, 2025a.

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

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

Etash Kumar Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, et al. Openthoughts: Data recipes for reasoning models. *CoRR*, abs/2506.04178, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *CoRR*, abs/2504.11456, 2025.

Dan Hendrycks, Steven Basart, Saurav Kadavath, Mantas Mazeika, Akul Arora, Ethan Guo, Collin Burns, Samir Puranik, Horace He, Dawn Song, and Jacob Steinhardt. Measuring coding challenge competence with APPS. In *Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

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

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL https://github.com/huggingface/open-r1.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. *CoRR*, abs/2403.07974, 2024.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

Yunjie Ji, Xiaoyu Tian, Sitong Zhao, Haotian Wang, Shuaiting Chen, Yiping Peng, Han Zhao, and Xiangang Li. Am-thinking-v1: Advancing the frontier of reasoning at 32b scale. *CoRR*, abs/2505.08311, 2025.

Jaehun Jung, Seungju Han, Ximing Lu, Skyler Hallinan, David Acuna, Shrimai Prabhumoye, Mostafa Patwary, Mohammad Shoeybi, Bryan Catanzaro, and Yejin Choi. Prismatic synthesis: Gradient-based data diversification boosts generalization in LLM reasoning. *CoRR*, abs/2505.20161, 2025.

Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, et al. Gemma 3 technical report. *CoRR*, abs/2503.19786, 2025.

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

Zhenyu Lei, Zhen Tan, Song Wang, Yaochen Zhu, Zihan Chen, Yushun Dong, and Jundong Li. Learning from diverse reasoning paths with routing and collaboration. *CoRR*, abs/2508.16861, 2025.

Hang Li, Kaiqi Yang, Yucheng Chu, Hui Liu, and Jiliang Tang. Exploring solution divergence and its effect on large language model problem solving. *CoRR*, abs/2509.22480, 2025a.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Numinamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf), 2024.

Rongao Li, Jie Fu, Bo-Wen Zhang, Tao Huang, Zhihong Sun, Chen Lyu, Guang Liu, Zhi Jin, and Ge Li. TACO: topics in algorithmic code generation dataset. *CoRR*, abs/2312.14852, 2023a.

Xingxuan Li, Yao Xiao, Dianwen Ng, Hai Ye, Yue Deng, Xiang Lin, Bin Wang, Zhanfeng Mo, Chong Zhang, Yueyi Zhang, Zonglin Yang, Ruilin Li, Lei Lei, Shihao Xu, Han Zhao, Weiling Chen, Feng Ji, and Lidong Bing. Miromind-m1: An open-source advancement in mathematical reasoning via context-aware multi-stage policy optimization. *CoRR*, abs/2507.14683, 2025b.

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

Yujia Li, David H. Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, Thomas Hubert, Peter Choy, Cyprien de Masson d'Autume, Igor Babuschkin, Xinyun Chen, Po-Sen Huang, Johannes Welbl, Sven Gowal, Alexey Cherepanov, James Molloy, Daniel J. Mankowitz, Esme Sutherland Robson, Pushmeet Kohli, Nando de Freitas, Koray Kavukcuoglu, and Oriol Vinyals. Competition-level code generation with alphacode. *CoRR*, abs/2203.07814, 2022.

Zheng Li, Xiang Li, Lingfeng Yang, Borui Zhao, Renjie Song, Lei Luo, Jun Li, and Jian Yang. Curriculum temperature for knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 1504–1512, 2023b.

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

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

Justus Mattern, Sami Jaghouar, Manveer Basra, Jannik Straube, Matthew Di Ferrante, Felix Gabriel, Jack Min Ong, Vincent Weisser, and Johannes Hagemann. Synthetic-1: Two million collaboratively generated reasoning traces from deepseek-r1, 2025. URL https://www.primeintellect.ai/blog/synthetic-1-release.

Mistral AI Team. Mistral 3. https://mistral.ai/news/mistral-3, December 2025. Accessed: 2025-12-03.

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

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

Nvidia. Openscience. [https://huggingface.co/datasets/nvidia/OpenScience](https://huggingface.co/datasets/nvidia/OpenScience), 2024.

NVIDIA. Nemotron 3 Nano: Open, efficient mixture-of-experts hybrid Mamba-Transformer model for Agentic reasoning, 2025. URL https://research.nvidia.com/labs/nemotron/files/NVIDIA-Nemotron-3-Nano-Technical-Report.pdf. Technical report.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. In *International Conference on Learning Representations*, 2016.

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

NovaSky Team. Sky-t1: Train your own o1 preview model within $450. https://novasky-ai.github.io/posts/sky-t1, 2025. Accessed: 2025-01-09.

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. Light-r1: Curriculum sft, DPO and RL for long COT from scratch and beyond. *CoRR*, abs/2503.10460, 2025.

Xiaojun Wu, Xiaoguang Jiang, Huiyang Li, Jucai Zhai, Dengfeng Liu, Qiaobo Hao, Huang Liu, Zhiguo Yang, Ji Xie, Ninglun Gu, Jin Yang, Kailai Zhang, Yelun Bao, and Jun Wang. Beyond scaling law: A data-efficient distillation framework for reasoning. *CoRR*, abs/2508.09883, 2025.

Jianzhi Yan, Le Liu, Youcheng Pan, Shiwei Chen, Yang Xiang, and Buzhou Tang. Towards efficient cot distillation: Self-guided rationale selector for better performance with fewer rationales. *CoRR*, abs/2509.23574, 2025.

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

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: less is more for reasoning. *CoRR*, abs/2502.03387, 2025.

Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, et al. ChatGLM: A family of large language models from GLM-130B to GLM-4 all tools. *CoRR*, abs/2406.12793, 2024.

Han Zhao, Haotian Wang, Yiping Peng, Sitong Zhao, Xiaoyu Tian, Shuaiting Chen, Yunjie Ji, and Xiangang Li. 1.4 million open-source distilled reasoning dataset to empower large language model training. *CoRR*, abs/2503.19633, 2025.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan L. Yuille, and Tao Kong. ibot: Image BERT pre-training with online tokenizer. *CoRR*, abs/2111.07832, 2021.