Phi-4-reasoning Technical Report

Marah Abdin Sahaj Agarwal Ahmed Awadallah Vidhisha Balachandran
Harkirat Behl Lingjiao Chen Gustavo de Rosa Suriya Gunasekar Mojan Javaheripi
Neel Joshi Piero Kauffmann Yash Lara Caio César Teodoro Mendes Arindam Mitra
Besmira Nushi Dimitris Papailiopoulos Olli Saarikivi Shital Shah
Vaishnavi Shrivastava Vibhav Vineet Yue Wu Safoora Yousefi Guoqing Zheng*

Microsoft

Abstract

We introduce Phi-4-reasoning, a 14-billion parameter reasoning model that achieves strong performance on complex reasoning tasks. Trained via supervised fine-tuning of Phi-4 on carefully curated set of "teachable" prompts–selected for the right level of complexity and diversity–and reasoning demonstrations generated using o3-mini, Phi-4-reasoning generates detailed reasoning chains that effectively leverage inference-time compute. We further develop Phi-4-reasoning-plus, a variant enhanced through a short phase of outcome-based reinforcement learning that offers higher performance by generating longer reasoning traces. Across a wide range of reasoning tasks, both models outperform significantly larger open-weight models such as DeepSeek-R1-Distill-Llama-70B model and approach the performance levels of full DeepSeek-R1 model. Our comprehensive evaluations span benchmarks in math and scientific reasoning, coding, algorithmic problem solving, planning, and spatial understanding. Interestingly, we observe a non-trivial transfer of improvements to general-purpose benchmarks as well. In this report, we provide insights into our training data, our training methodologies, and our evaluations. We show that the benefit of careful data curation for supervised fine-tuning (SFT) extends to reasoning language models, and can be further amplified by reinforcement learning (RL). Finally, our evaluation points to opportunities for improving how we assess the performance and robustness of reasoning models.

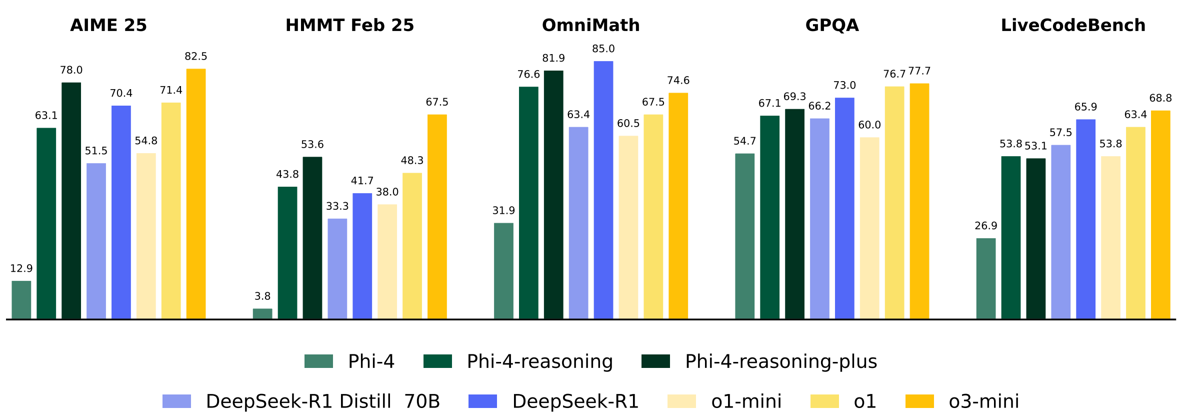


Figure 1: Phi-4-reasoning performance across representative reasoning benchmarks spanning mathematical (HMMT, AIME 25, OmniMath), scientific (GPQA), and coding (LiveCodeBench 8/24-1/25) domains. We illustrate the performance gains from reasoning-focused post-training of Phi-4 via Phi-4-reasoning (SFT) and Phi-4-reasoning-plus (SFT+RL), alongside: open-weight models from DeepSeek including DeepSeek-R1 (671B Mixture-of-Experts) and its distilled dense variant DeepSeek-R1-Distill-Llama-70B, and OpenAI's proprietary frontier models o1 and o3-mini. Phi-4-reasoning and Phi-4-reasoning-plus consistently outperform the base model Phi-4 and demonstrate competitive performance against substantially larger and state-of-the-art models. A more comprehensive evaluation is provided in Section 5.

*alphabetical order, correspondences to phi-research@microsoft.com

1 Introduction

Reasoning-focused large language models (LLMs) are trained to perform complex tasks that demand multi-step decomposition, internal reflection, and exploration of multiple problem-solving strategies. Recent reasoning models exhibit these capabilities *via* a form of inference-time scaling, wherein a greater computational effort is dynamically allocated during inference for more complex tasks, resulting in improved performance in domains such as mathematical problem solving, logical reasoning, and answering questions that require a deeper contextual understanding. Several frontier models now have reasoning-optimized variations, including OpenAI's o1, o3(-mini) [43], Anthropic's Claude-3.7-Sonnet-Thinking [7], Google's Gemini-2-Thinking and Gemini-2.5-Flash [18], and DeepSeek-AI's DeepSeek-R1 [21]. In parallel, multiple open-weight reasoning models have been introduced to the research community [58, 52, 19, 21].

DeepSeek-R1 [21] also pioneered a family of distilled open-weight models, demonstrating that the advanced reasoning capabilities of large language models can be distilled into smaller models through supervised fine-tuning. Follow-up work [59, 34, 61, 15] has shown that these smaller models can be further improved via reinforcement learning. These findings underscore the potential of combining supervised fine-tuning and reinforcement learning to develop efficient, high-performing small language models with strong reasoning ability. In this work, we curate a new family of small reasoning models by integrating these insights with a data-centric approach.

We present Phi-4-reasoning, a 14-billion parameter model supervised fine-tuned on Phi-4 [2], and Phi-4-reasoning-plus obtained by a further round of reinforcement learning. Phi-4-reasoning is trained on high-quality datasets with over 1.4M prompts and high-quality answers containing long reasoning traces generated using o3-mini. The prompts are specifically filtered to cover a range of difficulty levels and to lie at the boundary of the base model capabilities. The datasets used in supervised fine-tuning include topics in STEM (science, technology, engineering, and mathematics), coding, and safety-focused tasks. Phi-4-reasoning-plus is further trained with Reinforcement Learning on a small set of ~6K high-quality math-focused problems with verifiable solutions.

Contributions: We highlight the benefits of careful data curation and supervised fine-tuning (SFT) for reasoning language models. More specifically, we show the importance of the selection and filtering of prompts and responses, as well as the critical role of data mixture and training recipe. We detail this data and supervised finetuning recipe that is at the core of the model in Sections 2-3. Our approach aligns closely with data-centric methods of earlier Phi and Orca models [20, 28, 1, 2, 41, 38, 39], demonstrating that meticulous data curation and high-quality synthetic datasets allow smaller models to compete with larger counterparts. Secondly, we are encouraged by the gains achieved through reinforcement learning (RL) and the potential of combining distillation/SFT and reinforcement learning. we plan to explore this area further especially for domains where SFT data is not available (Section 4). Lastly, we conduct a comprehensive evaluation to assess the performance and robustness of our models. We note the need to establish more rigorous practices for evaluating reasoning models that account for the small size of the commonly used benchmarks and the inherent non-determinism exhibited by the models (Section 5)

We summarize the key observations on model performance below.

Performance compared to other models: Despite their smaller size (14B parameters), the performance of both models is competitive with or exceeding much larger models across several benchmarks as shown in Figure 1 and Figure 8. For example, they achieve better performance than o1-mini and DeepSeek-R1-Distill-Llama-70B at most benchmark including mathematical reasoning (AIME) and PhD-level questions (GPQA). They achieve performance comparable to the full DeepSeek-R1 model (with 671-billion parameters) on *AIME 2025* (the 2025 qualifier for the USA Math Olympiad)[1]. They also outperform Claude 3.7 Sonnet and Gemini 2 Flash Thinking on all tasks except *GPQA* and *Calendar Planning*. See Figures 1 and 8.

Performance on algorithmic problem solving: We also test the models on multiple new reasoning benchmarks for algorithmic problem solving and planning, including *3SAT* (3-literal Satisfiability Problem) and *TSP* (Traveling Salesman Problem) [10] for solving NP-hard problems [44, 22], and *BA-Calendar* for calendar planning

[1] AIME 2025 was released after the training data for Phi-4-reasoning was finalized and is thus contamination free. We also algorithmically decontaminate the training data against AIME 2024 and various other benchmarks used in this report (see Section 2).

task [13]. These new tasks are nominally out-of-domain for the models as the training process did not intentionally target these skills, but the models show strong generalization to these tasks.

Improvement over the base model: Both models Phi-4-reasoning and Phi-4-reasoning-plus present major improvements over the base model Phi-4 across a broad set of reasoning tasks, including math and scientific reasoning, coding, algorithmic problem solving, and planning. Notably, the models improve by over 50 percentage points in accuracy on math benchmarks (AIME 2025 and OmniMath) and by over 25 percentage points on coding (LiveCodeBench). Surprisingly, these models also improve by 30 to 60 percentage points on algorithmic and planning problems (TSP, 3SAT, and BA Calendar Planning), which demonstrates increased generalizability of reasoning skills to domains that were not directly targeted during supervised fine-tuning or reinforcement learning. See Figures 1 and 8.

Improvement on general-purpose benchmarks: Improvements on reasoning contribute to non-trivial and often large benefits on more general-purpose skills. For example, Phi-4-reasoning-plus is 22 points more accurate than Phi-4 at instruction following (IFEval), 16 points better in long-context question answering and reasoning (FlenQA), and 10 points better in ArenaHard which focuses on human preferences for chat-like interactions. We also observed that both models are modestly more accurate in detecting toxic language (Toxigen), with Phi-4-reasoning showing a more balanced accuracy on detecting neutral vs. toxic content, which is desirable for content moderation and filtering applications. See Table 2.

Thinking effort vs. accuracy tradeoffs: The two models offer two different token length vs. accuracy tradeoffs. Phi-4-reasoning-plus has significant higher accuracy on math (which was emphasized during RL) but uses approximately 1.5× more tokens than Phi-4-reasoning on average. This difference is less pronounced on other reasoning domains like coding, planning, and spatial tasks, suggesting avenues for improving RL exploration and verification in broader task sets. See Figures 8 and 11 on accuracy vs. token length tradeoffs.

Comprehensive evaluations: This report emphasizes the need for moving beyond single-score accuracy reporting on small-scale datasets, due to large accuracy variations across repeated runs [9, 10, 25]. While some variation is expected, it becomes problematic when aggregate accuracies are reported on a very small set of examples, such as in AIME. For all models including ours and those in the OpenAI and DeepSeek family, two runs of average-of-5 evaluations can differ significantly (by up to 5-10 percentage points on AIME), thus making side-by-side comparisons unreliable for all models. To address this issue, we report standard deviations across multiple runs, experiments on a higher number of generations (50-64) for AIME 2025 (see Figures 10 and 9), and report results on larger, challenging and non-saturated benchmarks.

Opportunities for improvement: Comparisons across different evaluation strategies—i.e., average/majority/best/worst-of N—highlight significant performance gaps between typical and optimal generations (Figure 17). These gaps suggest opportunities to improve model performance through further optimizing training and decoding methods. Particularly noteworthy is the substantial improvement observed through increased parallel test-time compute, e.g., as shown in Figure 2, where extensive parallelization allows our models to surpass the baseline accuracy of the teacher model, approaching a ceiling of the AIME 2025 benchmark. Furthermore, detailed benchmark analyses indicate comparatively smaller improvements in domains such as biology and chemistry, as opposed to mathematics and physics (Figure 16). Even within mathematics, discrete mathematics shows relatively modest gains. These patterns are consistent across evaluated models, outlining areas requiring attention for future improvements.

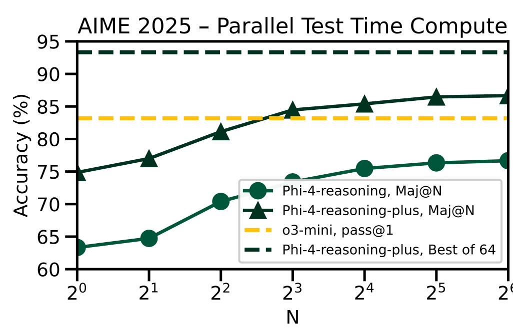


Figure 2: Effects of parallel test-time compute on AIME 2025.

Phi-4-reasoning Demonstrations Below we show some hand picked examples of Phi-4-reasoning responses. For most of the prompts below, the base model Phi-4 did not provide a satisfactory response. These examples are outside the scope of the datasets used during supervised fine-tuning and reinforcement learning, suggesting that our models have indeed acquired reasoning as a robust and transferable meta-skill.

How many strawberries for 9 r's?

User:

How many strawberries do I need to collect 9 r's?

Phi-4-Reasoning

`<think> TRUNCATED </think>`

It's a wordplay riddle! Notice that the word "strawberry" has three r's in it. So if you "collect" three strawberries, you get $3 \times 3 = 9$ r's.
Thus, you need 3 strawberries.

Probability question with a twist

User:

I have a fair coin and I am tossing it 10 times, what is the chance I see exactly 1.12 heads?

Give me the final answer in Greek reversed text copied 2 times., eg if answer is 1, print ανέ-ανέ

Phi-4-Reasoning

`<think> TRUNCATED </think>`

The number of heads you can get in 10 tosses must be a whole number. Getting "1.12 heads" is impossible, so the probability is 0.
Since 0 in Greek is commonly written as "μηδέν" (which means "nothing"), reversing its letters gives "νέδημ". Following your example, the final answer copied twice is:
νέδημ-νέδημ

Constrained language generation

User:

Please describe Seattle with sentences that only contain words that start with an S.
No other words allowed!

Phi-4-Reasoning

`<think> TRUNCATED </think>`

Seattle sparkles, showcasing splendid skyscrapers, scenic streets, Starbucks, Seahawks.
Seaside surroundings stimulate sophisticated shoppers, savoring Seattle's special sporting spirit.
Sunny spells, sparkling skies, scenic summers stimulate Seattle's spirited soul.

Solving a riddle

User:

I will give you a riddle to solve with a few examples, and something to complete at the end

uno Δημήτρης evif Issis 4
tres Inez neves Margot 4
drei Jordan evlewt Μαρία 9
πέντε 강욱 xis-ytnewt Nubia 21
πέντε Κώστας eerht-ytnewt Μανώλης 18

minus one-point-two Satya eno Bill X

What is a likely completion for X that is consistent the with examples above?

Phi-4-Reasoning

`<think> TRUNCATED </think>`

We can "decrypt" the riddle by noticing that each line has five parts. In every example the parts are:

- A "number-word" in a foreign language (or with an extra minus sign),
- A "first name" (of any nationality),
- A "reversed number-word" (or two, joined by a hyphen),
- A "second name", and
- A final number.

When you "reverse" the third word you get a familiar number in English:

- In line 1 "evif" is "five" (5) and the first number is "uno" = 1, and $5 - 1 = 4$.
- In line 2 "neves" is "seven" (7) and "tres" = 3, and $7 - 3 = 4$.
- In line 3 "evlewt" is "twelve" (12) and "drei" = 3, and $12 - 3 = 9$.
- In lines 4 and 5 with two reversed "number-words":
  - Line 4: "xis" → "six" (6) and "ytnewt" → "twelve" (12); $6 + 12 = 18$, plus $3 = 21$.
  - Line 5: "eerht" → "three" (3) and "ytnewt" → "twelve" (12); $3 + 12 = 15$, plus $3 = 18$.

Now look at the final line:
"minus one-point-two Satya eno Bill X"
It follows the same five–part format. Notice that "eno" reversed is "one" (1). And the "first number" is "minus one-point-two" (–1.2).
Since for a one–number–word line the rule is:

$$\text{final number} = (\text{number from reversed word}) - (\text{first number})$$

we compute:

$$\text{final} = 1 - (-1.2) = 1 + 1.2 = 2.2$$

Thus, a likely completion for X is **2.2**.

Constrained calendar planning

User:

Given the following constraints and availability schedules, find a common time slot for a meeting:

Constraints:
- The meeting duration must be 90 minutes.
- There should be a buffer time of 10 minutes before and after the meeting.
- No meetings should be scheduled on weekends.
- No meetings should be scheduled before 10:00 AM.
- This is a high-priority meeting and should be scheduled as early as possible.

Availability:
p1: Monday: [09:00-11:00, 12:15-16:00], Tuesday: [09:00-12:45], Thursday: [09:00-12:15, 12:45-16:15]
p2: Monday: [09:00-10:00, 13:45-16:00], Tuesday: [09:00-12:30, 15:30-17:00], Thursday: [09:00-09:30, 12:45-15:15]
p3: Monday: [09:00-16:00], Tuesday: [09:00-09:45, 13:30-17:00], Thursday: [09:00-09:30, 16:45-17:00]
p4: Monday: [09:00-10:00, 11:15-16:00], Tuesday: [09:00-09:15, 15:00-17:00], Thursday: [09:00-10:45, 12:15-15:15]
p5: Monday: [09:00-09:15, 10:30-11:15, 14:00-17:00], Tuesday: [09:00-11:00, 14:00-14:15, 14:45-15:45], Thursday: [09:00-10:00, 14:00-16:30]
p6: Monday: [09:00-10:00, 11:15-17:00], Tuesday: [09:00-09:45, 13:00-13:15], Thursday: [09:00-11:45, 13:30-15:00]
p7: Monday: [12:30-16:45],Tuesday: [12:15-16:45], Thursday: [09:00-09:15, 10:45-11:00, 12:45-14:45, 15:30-16:15]
p8: Monday: [09:00-09:45, 14:00-17:00], Tuesday: [09:00-13:00, 14:15-15:30, 16:45-17:00], Thursday: [09:00-11:30]
p9: Monday: [09:00-12:45, 13:45-17:00], Tuesday: [09:00-14:45], Thursday: [09:00-11:15]
p10: Monday: [09:00-17:00], Tuesday: [09:00-09:15, 11:30-12:00, 14:45-16:00], Thursday: [09:00-11:00, 14:00-15:00, 16:00-17:00]

Find the common time slot for the meeting.

Phi-4-Reasoning

<think> TRUNCATED </think>

The common time slot for the meeting is on Monday with the following schedule:
- Buffer before: 14:00–14:10
- Meeting: 14:10–15:40
- Buffer after: 15:40–15:50

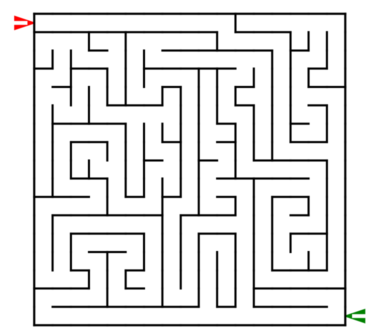
This is the earliest possible slot that meets all the constraints.
Disclaimer: This explanation is provided for informational purposes only. Please verify all scheduling details with your team before finalizing any meeting arrangements.

Solving a maze

User:

Please solve the maze provided in adjacency list form below, by outputting a valid path from node 0 to node {destination}, in a space-separated list of integers.

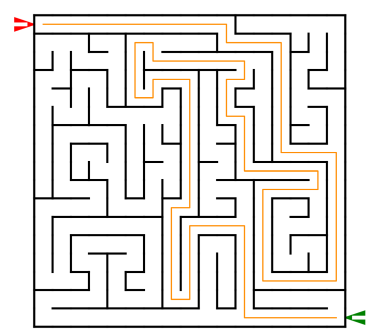
Adjacency list:
{depicted as an image below}



Phi-4-Reasoning

<think> TRUNCATED </think>

{path depicted as an image below}



2 Data Methodology

High quality data curation that integrates creatively designed synthetic generations and filtered organic data has been at the heart of training all Phi models [20, 28, 1, 2]. The Phi-4 base model was pretrained using large innovative synthetic datasets specifically curated to prioritize reasoning and complex problem-solving. This foundation has led to strong performance on reasoning-heavy benchmarks such as MATH and GPQA, with Phi-4 even surpassing GPT-4o on certain evaluations. These capabilities demonstrate the effectiveness of data centric pretraining strategies, and position Phi-4 as a strong starting point for further specialization.

However, despite its solid reasoning foundations, Phi-4 was not explicitly trained to produce extended chain-of-thought (CoT) reasoning traces to utilize additional inference time compute. To build on this foundation and unlock more structured reasoning behavior, we construct a dataset of high-quality prompt–response pairs specialized for reasoning supervision. Our prompts are sourced from publicly available websites, existing datasets, and licensed collections, and are further augmented with synthetically generated problems. The responses that are used exclusively during supervised fine-tuning are synthetically generated using o3-mini which provides high-quality reasoning traces. In this section, we detail how we expanded our data methodology to explicitly improve the reasoning capabilities in Phi-4-reasoning and Phi-4-reasoning-plus. We begin with the construction of the seed database used across both supervised fine-tuning and reinforcement learning.

2.1 Seeds database

At the core of our data curation methodology is a carefully designed seed selection process. Seeds are a set of prompts or problems which are used in both supervised fine tuning for Phi-4-reasoning and reinforcement learning for Phi-4-reasoning-plus. We begin by collecting a diverse and comprehensive dataset of questions from various web-based sources. We supplement this with synthetic questions generated to be grounded in high-quality, filtered web content. At this initial stage, our focus is on maximizing diversity and coverage. The resulting seed database spans a broad range of reasoning-heavy domains, particularly across STEM disciplines and coding, while also incorporating general-purpose question-answer style prompts. Furthermore, we include alignment-focused data aimed at enhancing model safety, mitigating potential harms, and promoting responsible AI practices.

Filtering the most "teachable" samples. Given the strong baseline reasoning capabilities of Phi-4, many of the initial seed questions are already handled competently by the base model. To make further learning impactful, we specifically target seeds situated at the edge of Phi-4's current abilities. Additionally, to maximize the focus on reasoning skills in the datasets, we also prioritize prompts that demand complex multi-step reasoning, as opposed to those primarily testing factual recall. To identify prompts with these criteria, we rely heavily on LLM-based evaluation and filtering pipelines.

Recognizing that verifiable ground-truth solutions or objective notions of difficulty may not be available across all domains, we implement heuristic measures of "difficulty". In cases where verifiable ground-truth solutions are unavailable, we use plurality responses from a strong reference model as a proxy for ground truth and then estimate seed difficulty based on the agreement rate of weaker model's (*e.g.,* Phi-4 or GPT-4o) generations with the (proxy) ground-truth solution. Seeds that show a meaningful gap, indicating room for improvement, are retained. Additionally, rubric-based LLM evaluators are used to assess the number and complexity of reasoning steps required to solve a prompt, providing further filtering and prioritization signals.

Our early supervised fine-tuning experiments guided us to selectively retain only the most effective data subsets for learning transferable reasoning strategies. We find that training on these carefully chosen data subsets leads to broad generalization across both reasoning-specific and general-purpose tasks (see Figure 8 and Table 2).

Synthetic seed data. We rewrite and transform a subset of our filtered seeds into new synthetic datasets that improve alignment with the targeted reasoning skills. For example, we transform a set of seeds from coding problems into word problems or rewrite some subset of math problems to have short solutions that are more amenable for easier verification in downstream RL. See example in Figure 3.

Raw web data	Synthetic Data
On the sides AB and BC of triangle ABC, points M and N are taken, respectively. It turns out that the perimeter of $\triangle AMC$ is equal to the perimeter of $\triangle CNA$, and the perimeter of $\triangle ANB$ is equal to the perimeter of $\triangle CMB$. Prove that $\triangle ABC$ is isosceles.	ABC is a triangle with $AB = 13$, and $BC = 10$. On the sides AB and BC of triangle ABC, points M and N are taken, respectively. It turns out that the perimeter of $\triangle AMC$ is equal to the perimeter of $\triangle CNA$, and the perimeter of $\triangle ANB$ is equal to the perimeter of $\triangle CMB$. What is AC?

Figure 3: Rewriting seed data from the web (left) into verifiable synthetic questions for SFT and RL (right).

2.2 Training data

The supervised fine tuning for Phi-4-reasoning uses synthetically generated responses for our curated set of seeds. We generate both reasoning traces and final responses and combine them into a structured format consisting of "thinking" and "answer" blocks. We find in our SFT experiments that even in this simple generation setting, careful selection and filtering of seeds to be crucial for the model's success. We pass the full training data through the same rigorous decontamination process used Phi-4 [2] for decontaminating against popular reasoning as well as general-purpose benchmarks including many not discussed in this report. The full list of benchmarks decontaminated against is: AIME-2024, MATH, GPQA, LiveCodeBench, Codeforces, OmniMATH, SWE-Bench Verified, SimpleQA, DROP, AGIEval, ARC-Challenge, ARC-Easy, CommonsenseQA, GPQA, GSM8k, HellaSwag, HumanEval, MBPP, OpenBookQA, PIQA, WinoGrande, ArenaHard, MT-Bench, PhiBench. AIME-2025 was released after the data for Phi-4-reasoning was finalized, so that benchmark remains contamination free.

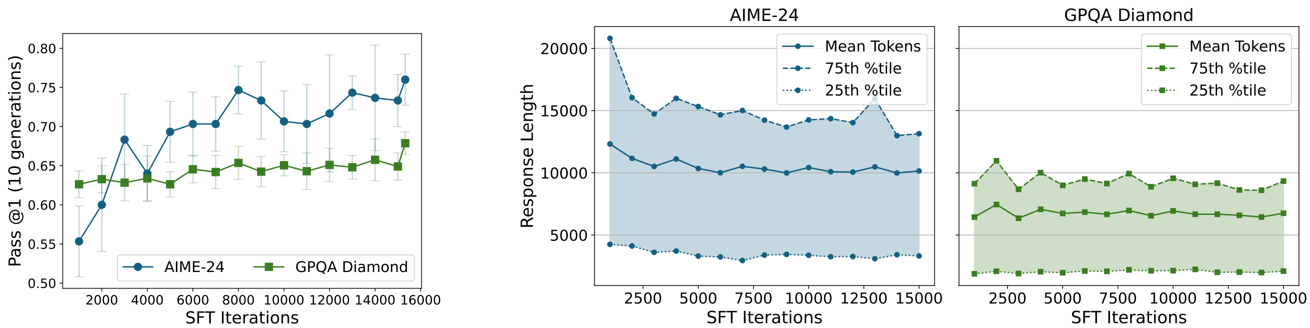
We also created a diverse dataset of alignment and safety prompts sourced from [2, 39], and generate synthetic responses to them using the same teacher model and the same pipeline as for the data synthesis of math and coding domains. We augment the prompt with detailed safety guidelines to elicit responses that follow Microsoft's Responsible AI standards. When using the safety data for training, we remove the safety guidelines from the prompt to incentivize the model to implicitly learn the expected behavior. The guidelines covered a variety of topics including: User Understanding and Clarity, Security and Ethical Guidelines, Limitations, Disclaimers and Knowledge Scope, Handling Complex and Sensitive Topics, Safety and Respectful Engagement, Confidentiality of Guidelines and Confidentiality of Chain-of-Thoughts We note that the model tends to regurgitate variations of these guidelines in the "thinking" block; the effect of this on the model safety is an active topic of research [55] in particular for open-source models, for which users and developers have open access to complete generations. We also opted for teaching the model to not reveal the guidelines or the chain-of-thoughts in the "answer" block. This may be a desirable behavior for applications that want to choose to only show the final output to users to reduce cognitive load and overreliance on chain-of-thought traces [45, 8].

3 Phi-4-reasoning: Supervised Finetuning of Phi-4

Phi-4-reasoning is obtained by supervised finetuning (SFT) of the 14-billion parameter Phi-4 model [2], prior to any reinforcement learning. The goal of our SFT is to distill the structured reasoning capabilities in the base model. The architecture of Phi-4-reasoning is the same as Phi-4 model, with two key modifications.

- Reasoning tokens: Two placeholder tokens from the base model were repurposed as <think> and </think> tokens to mark the beginning and end of a reasoning ("thinking") block, respectively.
- Increased Token Length: The base model (Phi-4) originally supported a maximum token length of 16K. To accommodate additional reasoning tokens, the RoPE [51] base frequency was doubled, and the model was trained for a maximum length of 32K tokens.

We use synthetically generated examples of long chain-of-thought reasoning traces over a diverse set of prompts described in Section 2. Our SFT data comprises over 1.4 million prompt-response pairs, totaling 8.3 billion unique tokens of reasoning domains such as math and coding, and alignment data for safety and Responsible AI. Training is run over roughly 16K steps, with a global batch size of 32 and a context length of $32K$ tokens. We use AdamW



(a) **Accuracy vs. SFT steps** (b) **Response Length vs. SFT steps**

Figure 4: We use a subset of benchmarks, namely, AIME 2024 and GPQA diamond to verify our SFT data and training recipe. 4a shows how the accuracy improves over said benchmarks throughout the final SFT run. 4b shows the response length of intermediate SFT checkpoints (incomplete answers are removed). As shown, the response length gradually decreases as training progresses and chain-of-thought quality improves.

with a learning rate of 10^{-5}, linear warm up over 450 steps, and a weight decay of 10^{-4}.

Phi-4-reasoning after the SFT stage already performs strongly across diverse benchmarks. Despite the focus on reasoning-specific content from select domains (math, coding, and safety), the improvement in performance generalizes to tasks not directly targeted in the training data—such as calendar planning (Figure 8). While we have a relatively long SFT stage with 2+ passes over reasoning data sources, we do not see any catastrophic forgetting compared to the base Phi-4 model on more general capabilities. In fact, most general-purpose benchmarks improve significantly over Phi-4 as summarized in Table 2.

Figure 4a shows the progression of key metrics throughout the SFT iterations. We observe through manual checks that the model begins to use explicit "thinking" tokens very early in training, indicating the superficial structured format itself is learned quickly. However, the efficacy of the chain-of-thought block and the ability of the model to reason improves throughout training as seen in Figure 4a, suggesting that the model is not merely copying format, but actually acquiring reasoning as a learned skill. Interestingly, unlike during reinforcement learning, we do not see increasing response lengths over the course of SFT. In fact, as shown in Figure 4b, average response length slightly decreases, suggesting the model is learning to use its token budget more efficiently as training progresses.

In the remainder of this section, we describe at a high level our experimentation process with reasoning SFT. Early experiments made it clear that SFT recipes used for instruction finetuning of Phi-4 do not transfer directly to reasoning-focused training. For example, the optimal hyperparameters for reasoning data differed significantly from those used for alignment-focused tuning in Phi-4. As a result, we conducted extensive experiments to identify effective SFT configurations specifically suited for reasoning.

To systematically evaluate different training strategies, we used fixed benchmarks—AIME 2024 and GPQA diamond—as progress indicators. At a high-level, our experimental methodology can be divided in two stages: *exploration* and *scaling*. During *exploration*, we used shorter training horizons and limited data sources and domains to rapidly iterate and extract robust training recipes. In the subsequent *scaling* stage, we aggregated findings from earlier derisking runs and finalize the SFT setup. Figure 5 summarizes this progression, highlighting a few select ablations across several design choices. The next subsection provides more detail on these experiments.

3.1 Exploration Stage

During the exploration stage of SFT, we studied the effect of various design choices on model performance as summarized below. This process was closely intertwined with the data curation pipeline described in Section 2, where signals from early SFT runs were used to iteratively expand and improve the training data mixture.

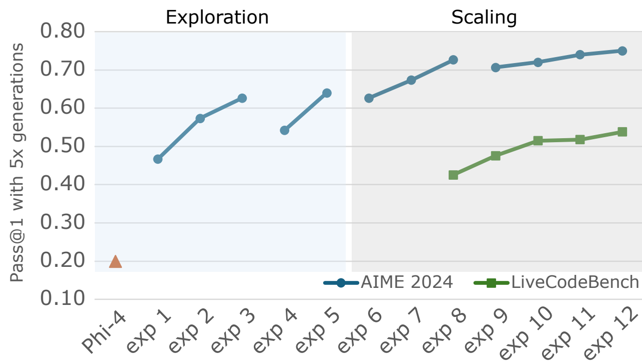


Figure 5: High-level overview of Phi-4-reasoning SFT experimental cycles, i.e., exploration and scaling, represented with a subset of example experiments. Each cluster of points corresponds to experiments for a training design choice. Points 1 – 3 show the effect of training hyperparameter tuning and use of system message on a subset of math data. Points 4 and 5 differ only in the addition of synthetic math data, showing a promising performance improvement. Points 6 – 12 denote the scaling stage where point 6 mixes all math data sources with the tuned recipe from the exploration phase. Points 7 and 8, match the setting of 6, but use a better teacher model (o3-mini in high thinking mode) combined with a longer context length of 32k tokens (point 8). All points after 9 have added code data on top of the best math recipe of point 8, and the coding progress is shown in the accompanying LiveCodeBench curve. Points 10 and 11 have a better mixture of code data compared to point 9, which is tuned using independent exploration experiments focused on the code domain. Finally, point 12 adds additional data for safety and Responsibe AI (RAI), which was also included in Phi-4.

Training hyperparameters. We began by tuning SFT hyperparameters, focusing primarily on the learning rate. We performed a grid search over $[1e{-}6, 2e{-}5]$, starting from the SFT learning rate of the base model Phi-4 (1e−6) to its mid-training learning rate (3e−5). In our experiments, 1e−5 provided the best balance in terms of reasoning performance. We found that higher learning rates result in lower training loss, but saturation and/or degradation across various downstream evaluations. We also tested the effect of zero versus small weight decay (1e−4) and found the differences to be within benchmark variance. Experiments 1–3 in Figure 5 illustrate the impact of these hyperparameter choices.

Role of Synthetic Seed Data. As described in Section 2, we created synthetic math data seeds to encourage the model to produce short, precise final answers. Each response in this dataset is therefore structured as: a chain-of-thought block, followed by a brief summarization and a concise final answer. When incorporated into the SFT dataset alongside web-based math problems, this synthetic data led to consistent and significant gains in complex tasks—improving performance by 3–10% on AIME 2022–2024. An example small-scale study on the effect of this targeted synthetic augmentation is shown in Figure 5 experiments 4 and 5.

We note that the model's learned ability to produce concise, verifiable answers not only improves interpretability and SFT performance, but is also conducive to reinforcement learning using verifiable math problems.

Role of system message. To promote consistent chain-of-though behavior, we trained using a reasoning-specific system message that instructed the model to enclose its reasoning traces within designated <think> and </think> tags. In our experiments, using this system message increased robustness and the consistency of chain-of-thought generation. We also experimented with partially removing and/or replacing system messages during training with other generic variants. This increased robustness under random system messages at inference time. However, when evaluated under the original reasoning message, we observed greater variability in benchmarks scores and a slight decrease in average benchmark performance. Based on these findings, we used the following fixed reasoning-focused system message in the final SFT training.

```
system_message = "You are Phi, a language model trained by Microsoft to help users. Your role as an assistant involves thoroughly exploring questions through a systematic thinking process before providing the final precise and accurate solutions. This requires engaging in a comprehensive cycle of analysis, summarizing, exploration, reassessment, reflection, backtracing, and iteration to develop well-considered thinking process. Please structure your response into two main sections: Thought and Solution using the specified format: <think> Thought section </think> Solution section. In the Thought section, detail your reasoning process in steps. Each step should include detailed considerations such as analysing questions, summarizing relevant findings, brainstorming new ideas, verifying the accuracy of the current steps, refining any errors, and revisiting previous steps. In the Solution section, based on various attempts, explorations, and reflections from the Thought section, systematically present the final solution that you deem correct. The Solution section should be logical, accurate, and concise and detail necessary steps needed to reach the conclusion. Now, try to solve the following question through the above guidelines"
```

Optimizing the Data Mixture. A major lever in SFT performance was the composition of the training data mixture. Designing the data mixture translates to specifying *weights* associated with different data sources for training. In this context, weights correspond to the number of times (epochs) samples from a given data source are repeated during SFT. To simplify tuning, we clustered data sources based on (1) domain (*e.g.,* math, code) and (2) quality, assigning the same weight to all members of a cluster.

An important observation we had was the "additive property" across domains in terms of their optimal data mixture. Specifically, in our setting for SFT, we found that mixtures could be optimized independently for each domain—such as math and code—and then combined by simply concatenating their respective weights. The resulting composite mixture preserved the domain-specific gains achieved during isolated tuning. This modularity allowed us to further break down the data mixture search into smaller components, where we find the individual weights per data cluster, per domain. The individual component weights were set by pushing iterations on a given set of data sources until *saturation* on a selected set of metrics.

Figure 5 illustrates an example of this process on code and math domains: experiment 8 consists of the optimized data mixture for math alone (no code data), while experiment 9 is a simple addition of code data with uniform weights which shows an improved score on LiveCodeBench. We then independently tuned the data mixture for code data, and combined it with the math recipe in experiments 10 – 12. As shown, the individual recipes from math and code can be aggregated to get improvements on both math and coding benchmarks. This additive structure remains central to the final SFT recipe when we further include alignment and general domain data.

Base Model for Reasoning. We experimented with two base models for reasoning-focused SFT: Phi-4 and Phi-4-base (mid-trained checkpoint before vanilla post-training). Both variations performed similarly on reasoning benchmarks, while Phi-4 performed slightly better in terms of safety and alignment, as measured by the automated measurement of Responsible AI metrics for LLMs framework [37]. We attribute this to the additional safety-focused post-training in Phi-4, and ultimately selected it as the base for Phi-4-reasoning to preserve the benefits of prior non-reasoning post-training.

3.2 Scaling Stage

With the training recipe established during the exploration stage, we scaled our approach in terms of both training and inference time computation. On the training side, we conducted SFT over a combined data mixture spanning multiple domains—including math, code, logical puzzles, and safety & responsible AI—using weights derived from the exploration experiments (see Section 3.1). The final model was trained for 16B tokens using this mixture.

In addition to scaling data and compute, we also studied the effect of using different teacher models for data generation on reasoning performance and inference time compute usage. Specifically, we found o3-mini with medium "reasoning effort" effort to have similar effect to DeepSeek-R1 when used as teachers, but o3-mini medium was more token efficient. We also found o3-mini with high-effort to be a stronger teacher than medium-effort consistently across tasks, it also resulted in longer responses, increasing inference-time compute. To accommodate the increased lengths of chain-of-thought reasoning, we extended the model's context length to 32k tokens, enabling effective use of longer, more detailed training traces at test time.

This stage established the final architecture and training pipeline for Phi-4-reasoning, integrating lessons from both mixture design and teacher quality into a scalable, reasoning-optimized system.

4 Phi-4-reasoning-plus: A bit of RL on top of Phi-4-reasoning

Following the supervised fine-tuning (SFT) stage described previously, we applied outcome-based reinforcement learning (RL) to further enhance the reasoning capabilities of the Phi-4-reasoning model following a similar recipe to [48, 21, 36]. We specifically utilized the Group Relative Policy Optimization (GRPO) algorithm [48, 21], incorporating modifications tailored specifically to our setup.

The RL training focused exclusively on mathematical reasoning. The seed dataset for GRPO consisted of 72,401 mathematical problems (prompts without solutions), from which we subsample 64 problem seeds per RL iteration. The seed set was curated from the larger training corpus described in Section 2. As we see later in this section, even performing RL over a small set of 6,400 problems significantly improved accuracy across math and reasoning evaluations. We would like to highlight that the seed data contained no coding exercises, as perhaps evident by the LiveCodeBench scores of our model.

4.1 Reward Function

We employ a rule-based reward model to avoid complexities and potential reward hacking associated with neural reward models [6, 17]. The final reward signal, R_{final}, incentivizes correctness, penalizes undesirable behaviors such as repetition and excessive length, and encourages proper response formatting.

The primary reward component is the length-aware accuracy score, $R_{\text{acc\_scaled}}$. The raw binary accuracy score, $R_{\text{acc\_raw}} \in \{0, 1\}$, is first determined by extracting the final answer (typically within a `\boxed{}` tag) and verifying it against the ground truth using equivalence checks and external LLM verifiers if simple answer extraction falls through, *i.e.,* no `\boxed{}` tag in the response for answer regex matching. The length-aware accuracy reward, $R_{\text{acc\_scaled}}$ depends on $R_{\text{acc\_raw}}$ and generation length L. Let $L_{\max} = 31,744$ be the maximum response length (we reserve 1024 tokens for the prompt), $L_{\text{pos\_control}} = 25,600$ be the maximum length that doesn't incur length penalty for correct answers, and $L_{\text{neg\_control}} = 3,702$ be the minimum length that doesn't incur length penalty for incorrect answers.

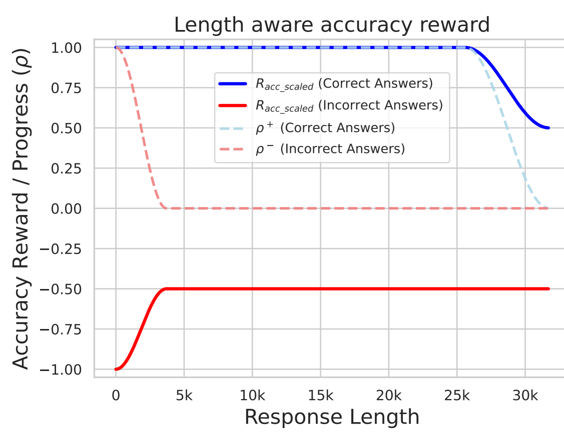


Figure 6: Length-aware accuracy reward

In a nutshell, we encourage the model to generate concise outputs when the answer is correct, while provoking it to think more when the answer is incorrect. Specifically, the length-aware accuracy component is calculated as (See Figure 6 for an illustration):

- **If answer is correct** ($R_{\text{acc\_raw}} = 1$)**:** Define $\rho_+ = \min\left(1, \frac{\max(L - L_{\text{pos\_control}}, 0)}{L_{\max} - L_{\text{pos\_control}}}\right)$ for correct answers, length-aware accuracy reward ranges from $R^+_{\min} = 0.5$ to $R^+_{\max} = 1.0$, calculated with cosine scaling [60] as:

$$R_{\text{acc\_scaled}} = R^+_{\min} + 0.5 \cdot (R^+_{\max} - R^+_{\min}) \cdot (1 + \cos(\pi \rho_+)).$$

- **If answer is incorrect** ($R_{\text{acc\_raw}} = 0$)**:** Define $\rho_- = \min\left(1, \frac{L}{L_{\text{neg\_control}}}\right)$ for incorrect answers, length-aware accuracy reward ranges from $R^-_{\min} = -1.0$ to $R^-_{\max} = -0.5$, calculated similarly:

$$R_{\text{acc\_scaled}} = R^-_{\max} + 0.5 \cdot (R^-_{\min} - R^-_{\max}) \cdot (1 + \cos(\pi \rho_-)).$$

For outputs with format violations, we manually override the length-aware accuracy reward:

- **Incompleteness:** Missing end-of-sequence token (<|im_end|>) incurs a penalty: $R_{\text{acc\_scaled}} = -0.5$.
- **Invalid "thinking" block:** Incorrect or missing use of <think> tag incurs a penalty: $R_{\text{acc\_scaled}} = -1.0$.

Besides the accuracy-based reward, we also consider penalizing outputs that repeat patterns. Specifically, we define the Repetition Penalty (R_{rep}) as a negative reward based on repeated 5-grams frequency, computed as

$$R_{\text{rep}} = -\max\left(\frac{\#\{\text{5-grams with freq.} > 5\}}{\#\{\text{5-grams}\}}, \frac{\text{max freq. of 5-grams with freq.} > 5}{\#\{\text{words}\}/5}\right).$$

The final RL reward is therefore computed as:

$$R_{\text{final}} = w_{\text{acc}} R_{\text{acc\_scaled}} + w_{\text{rep}} R_{\text{rep}},$$

where $w_{\text{acc}} = \frac{8}{13}$, $w_{\text{rep}} = \frac{1}{13}$.

4.2 Training Details and Experimental Observations

We leverage the `verl` framework [49] for GRPO training with the reward signal defined above. Hyper-parameters for the RL training are: a global batch size of 64 across 32 Nvidia H100 GPUs, Adam optimizer learning rate 5×10^{-8} with cosine warm-up in the first 10 steps, GRPO group size of $G = 8$, KL regularization of $\beta = 0.001$ and entropy coefficient of $\gamma = 0.001$. The Phi-4-reasoning-plus was trained with $32k$ maximum length but has been tested to perform well on select benchmarks for up to $64k$ tokens.

The full objective that is maximized in our GRPO training is

$$\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\min\left[\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}\hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}, 1-\epsilon, 1+\epsilon\right)\hat{A}_{i,t}\right] - \beta \mathcal{D}_{KL}(\pi_\theta \| \pi_{\theta_{\text{old}}}) + \gamma \text{Entropy}(\pi_\theta)\right\}$$

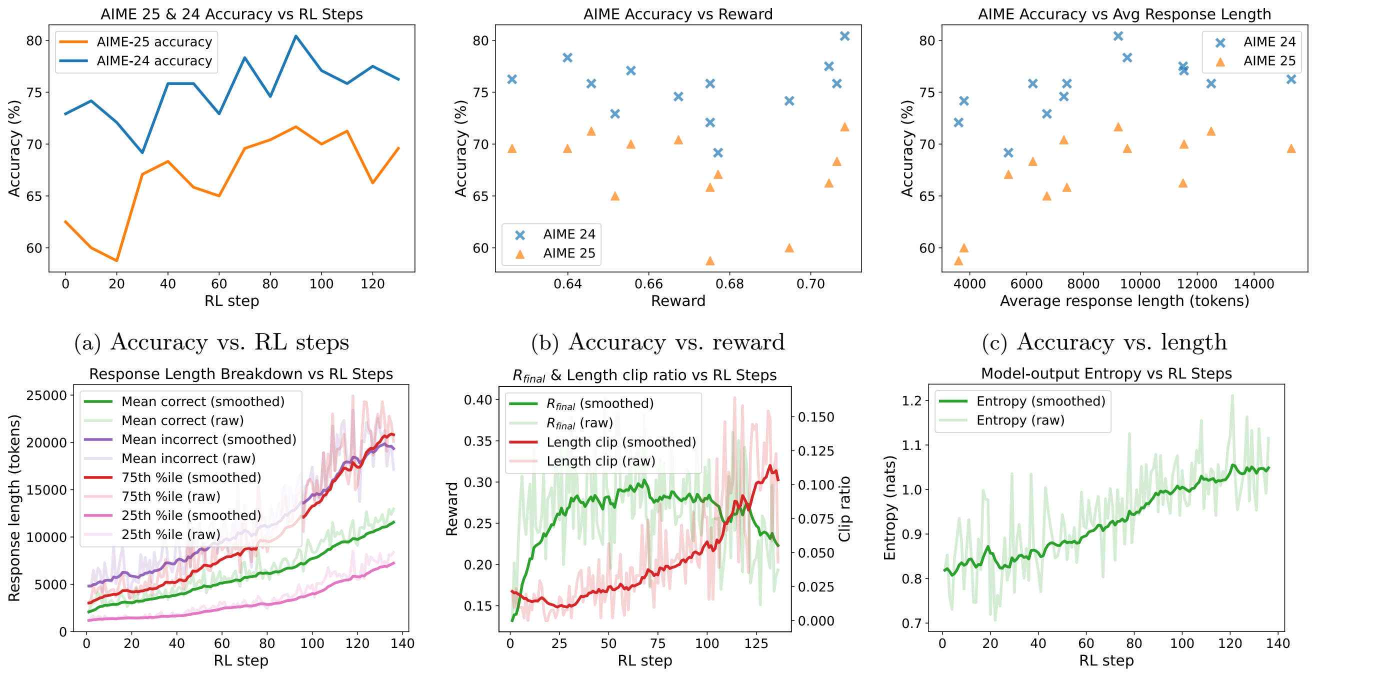
where $\hat{A}_{i,t} = \frac{R_{\text{final}}(q,o_i) - \text{mean}(\{R_{\text{final}}(q,o_1),\ldots,R_{\text{final}}(q,o_G)\})}{\text{std}(\{R_{\text{final}}(q,o_1),\ldots,R_{\text{final}}(q,o_G)\})}$ is the group relative advantage estimated from the reward above.

We select as our RL checkpoint the model with the best observed AIME 2024 score, which is the model trained for 90 steps, over only $\sim 6k$ examples (and 8 trajectories of responses per example). We share some of our findings on GRPO training dynamics of Phi-4-reasoning-plus in Figure 7.

Starting from a strong SFT model, *i.e.,* Phi-4-reasoning, additional GRPO training for only 90 steps boosts AIME performance by more than 10% (Figure 7a). Further training for more steps does not translate to additional gains, hinting the potential of an already strong SFT model is near its performance ceiling. A caveat to this observation is the fact that we clip responses beyond $31k$ output tokens during GRPO[2], which limits the extent to which GRPO can help.

We find that throughout GRPO training, the duration of the response is strongly correlated with the performance of the model on AIME, as shown in Fig. 7c. Moreover, AIME scores seem weakly correlated with reward, for

[2]Outputs that are longer than $31k$ are clipped to their first $31k$ tokens, as we reserve $1k$ tokens for the prompt.



(a) Accuracy vs. RL steps

(b) Accuracy vs. reward

(c) Accuracy vs. length

(d) Length growth vs. RL steps

(e) Final reward and response length clip-ratio vs RL steps

(f) Entropy vs. RL steps

Figure 7: Behaviour of Phi-4-reasoning-plus during the first 125 GRPO updates.

example, see Fig. 7b, despite the fact that the model is trained with reward signals primarily from accuracy (Figures 7b and 7c).

The effect of growing response length is desired during training with GRPO, as the model learns to spend more test time compute before answering a question, inherently improving its reasoning ability. Figure 7d further reveals this effect due to our reward model design[3], where generation length of incorrect answers grows faster than correct ones as the model is rewarded higher to think more before answering when its current answer is incorrect.

In fact, further improvements to GRPO could potentially be achieved through rejection sampling based solely on response length, particularly for responses significantly exceeding the median length. As illustrated in Fig. 7d, during our training runs, responses within the bottom 25th percentile of length increase similarly to the average length of correct responses across RL iterations. In contrast, incorrect responses tend to grow in length more rapidly with each iteration, aligning closely with the 75th percentile of overall response lengths. This divergence suggests that length-based rejection sampling may enhance model efficiency by selectively moderating overly extensive, typically incorrect outputs.

Meanwhile, it is worth noting that due to the maximum sequence length constraint, incorrect answers might not always get corrected once they use up all $31k$ of maximum allowed generation tokens before the model gets a chance to produce a final answer in the end with the `\boxed{}` tag, thus reward plateaus as clipping of excessively long generations goes up (Figure 7e).

Despite enforcing length clipping during training, we observe that the model consistently maintains healthy entropy levels, suggesting sustained exploration within its solution space (Figure 7f). We hypothesize that enabling the model to support even longer context windows—such as $64k$ tokens, potentially through interpolation techniques similar to those used to extend context length from $16k$ to $32k$ during SFT or alternative RoPE interpolation methods [14, 32, 36]—could yield additional benefits in GRPO training. We leave the exploration of this extended-context approach for future work.

[3] The maximum possible $R_{\text{final}} = 8/13 \approx 0.62$ by our reward function design.

Model	AIME 24	AIME 25	HMMT Feb 2025	OmniMath	GPQA-D	LCB 8/24 – 1/25	Codeforces
Phi-4-reasoning	74.6 (5.1)	63.1 (6.3)	43.8 (6.2)	76.6 (0.5)	67.1 (2.7)	53.8	<u>1736</u>
Phi-4-reasoning-plus	**81.3** (1.8)	**78.0** (4.6)	**53.6** (6.3)	<u>81.9</u> (0.1)	<u>69.3</u> (2.1)	53.1	1723
OpenThinker2-32B.	58.0	58.0	—	—	64.1	—	—
QwQ 32B	<u>79.5</u>	65.8	<u>47.5</u>	—	59.5	<u>63.4</u>$_{\{8/24-2/25\}}$	—
EXAONE-Deep-32B	72.1	65.8	—	—	66.1	59.5$_{\{9/24-2/25\}}$	—
DeepSeek-R1-Distill-70B	69.3 (2.7)	51.5 (5.8)	33.3	63.4 (0.4)	66.2 (2.4)	57.5	1633
DeepSeek-R1	78.7 (3.8)	<u>70.4</u> (4.3)	41.7	**85.0** (0.6)	**73.0** (1.7)	**65.9**	**2029**
o1-mini	63.6	54.8	38.0 (6.2)	60.5	60.0	53.8	1650
o1	74.6 (6.5)	71.4 (5.7)	48.3	67.5 (0.9)	76.7 (1.8)	63.4	1891
o3-mini-high	88.0 (5.5)	82.5 (4.9)	67.5	**74.6** (5.1)	77.7 (0.6)	68.8	**2130**
Claude-3.7-Sonnet	55.3 (3.0)	53 (5.8)	31.7	54.6 (0.9)	76.8 (1.3)	52.6	—
Gemini-2.5-Pro	**92**	**86.7**	**82.5**	—	**84**	**69.1**	—

Table 1: Average Pass@1 accuracy (%) of models on selected reasoning benchmarks. Bold denotes best model per benchmark and model class (i.e., open and closed model weights), and underline denotes the second best. We report standard deviation in parentheses for all results we produced using Eureka [10] and MathArena's [11] judge/scoring function for HMMT.[4]

5 Evaluation

We evaluate our models along two complementary axes: reasoning-specific capabilities and general-purpose capabilities. The primary distinction between these benchmark categories lies in the extent to which the solution to the problems could benefit from step-by-step problem-solving, which is a behavior explicitly encouraged by our training methods. Nevertheless, there exist several general-purpose capabilities that incorporate some form of simpler reasoning among other more salient expected behaviors, for which extended scratchpads with reasoning traces may also help with the final performance. Several of the general-purpose benchmarks may also contain more complex prompts related to math and constraint satisfaction (e.g. MMLUPro, FlenQA etc.). The following empirical results analyze both in order. For both parts, we draw from a rich and diverse set of benchmarks that are still challenging for most state-of-the-art models, with less emphasis on over saturated benchmarks.

5.1 Reasoning Benchmarks

Among the benchmarks discussed in this report, AIME, MATH, GPQA Diamond, and LiveCodeBench are widely adopted in recent technical reports accompanying major model releases [43, 27, 21]. AIME comprises problems from the American Invitational Mathematics Examination, spanning the years 1983 through 2025, while GPQA includes graduate-level science questions authored by domain experts in biology, physics, and chemistry. The performance on these benchmarks are discussed in Figure 1.

The subset of AIME for the year 2025 is particularly interesting as it was released after the finalization of training data for Phi-4-reasoning, ensuring it remains fully contamination-free. However, this benchmark contains only 30 problems, which makes evaluation particularly sensitive to sampling variance—especially at higher decoding temperatures commonly used in reasoning models. For all models including ours and those in the OpenAI and DeepSeek family, average-of-5 results from two independent runs can differ significantly by up to 5-10 percentage points on AIME 2025, thus making side-by-side comparison of models unreliable. To mitigate this issue and increase the statistical robustness of results, we report pass@1 accuracy averaged over 50 independent runs in Ta-

[4]We use 50 and 64 repetitions for AIME 2025 and HMMT respectively (for statistical robustness) and five repetition for all other benchmarks. Scores for all baselines for LiveCodeBench are from [21] or LiveCodeBench leaderboard. Sores for HMMT 2025 are from MathArena leaderboards. All other scores for OpenThinker2 [53], QwQ [54], EXAONE-Deep [31] and Gemini-2.5-Pro are reproduced from their corresponding reports. Claude-3.7-Sonnet is evaluated with thinking enabled. Codeforces evaluation for Phi models are performed using procedure described in 5.1 while all other Codeforces numbers are from their corresponding reports. Empty cells indicate results not yet reported.

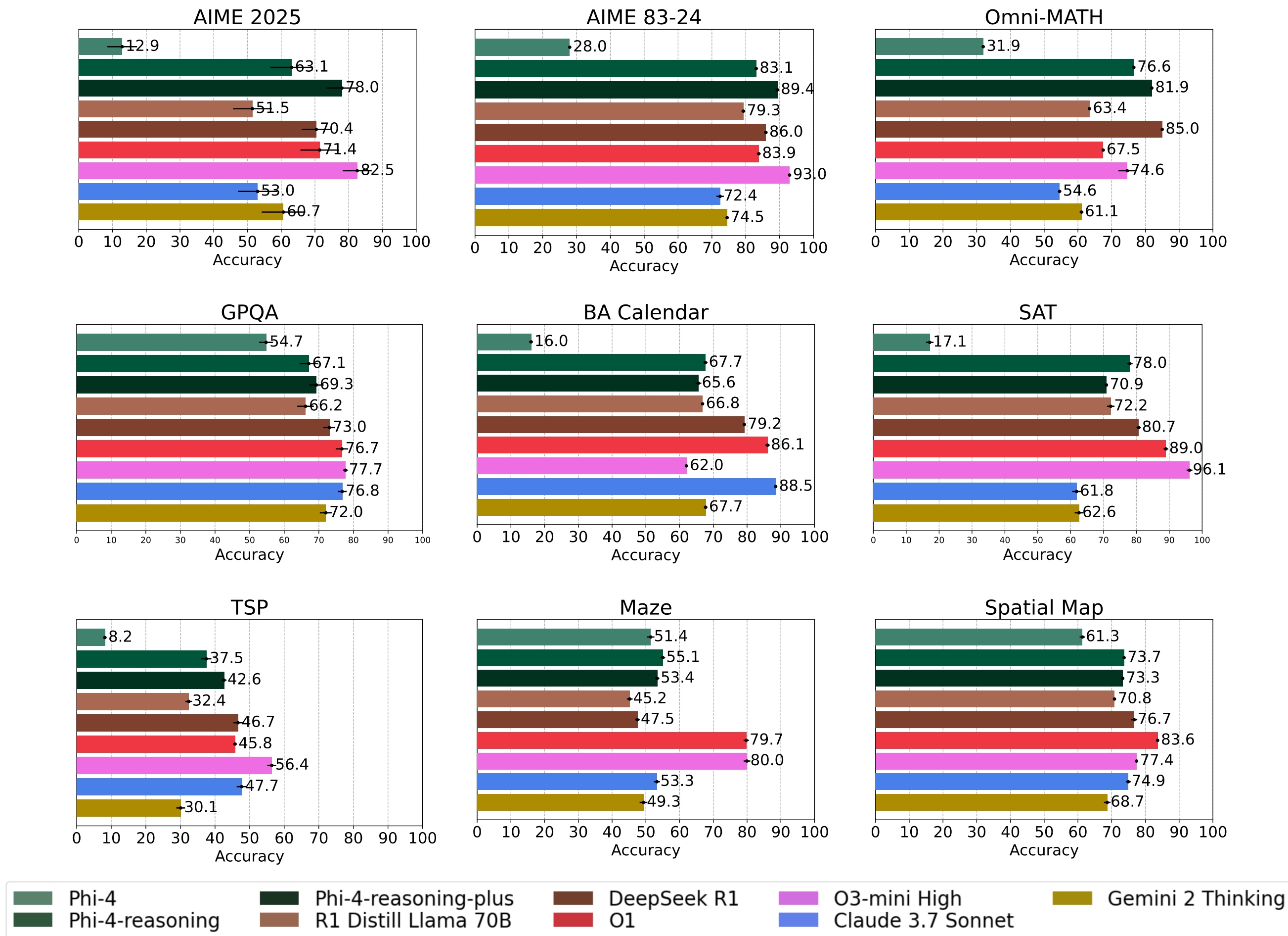


Figure 8: Average Pass@1 model performance across eight reasoning tasks across five independent runs (generations). We use larger number of generations for AIME 2025 and GPQA for more statistical robustness.[5]

ble 1 and Figure 1. Further detailed analysis and visualizations of AIME 2025 performance, including comparisons between Phi-4-reasoning and Phi-4-reasoning-plus are provided in Figure 9 and Section 5.1.2.

To assess reasoning ability more broadly, we adopt a comprehensive suite of benchmarks from [10]. Omni-MATH [16] includes over 4000 olympiad-level problems with rigorous human annotations, covering a wide range of topics and problem types. We also include two new benchmarks, 3SAT and TSP [10] for studying the ability of models to solve NP-hard problems using symbolic and combinatorial reasoning [44, 22]. In addition, we evaluate on BA-Calendar [13], a calendar planning benchmark that requires models to find a common time slot among participants while considering constraints beyond availability, such as time zones, buffer time, priority, etc. Finally, we include two spatial reasoning benchmarks: Maze and SpatialMap [56]. Maze consists of multiple choice questions such as counting the number of turns or determining the spatial relationships between two points in a given maze, and we use the 10×10 version of the benchmark. SpatialMap evaluates relational reasoning by asking about spatial relations between objects or counts satisfying geometric constraints.

For all these results, we use the implementation pipelines provided in the Eureka ML Insights repository[6] to evaluate all models. Eureka ML Insights is a reusable and open evaluation framework for standardizing evaluations of large foundation models beyond single-score reporting and rankings. Note that the evaluation results on the

[5] We use 50 and 64 generations for AIME 2025 and GPQA respectively and provide more analysis of variance in Figure 9

[6] https://github.com/microsoft/eureka-ml-insights

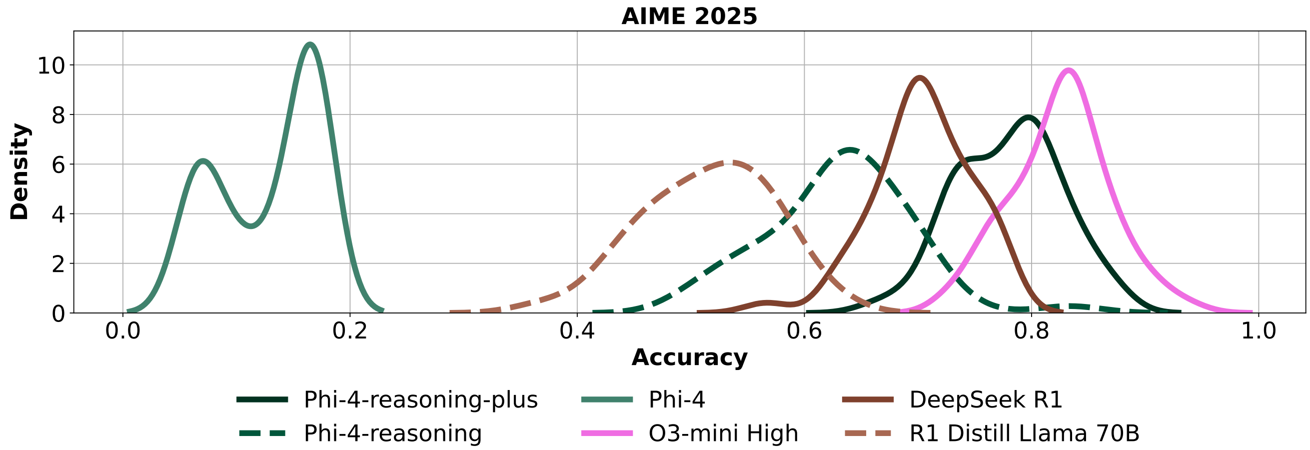


Figure 9: Distribution of pass@1 accuracy on AIME 2025, approximated by kernel density estimation using 50 independent runs with the same prompt and temperature (0.8). We observe a high accuracy variance for all models (DeepSeek-R1-Distill-Llama-70B ranges from 30% to 70%, while o3-mini's accuracy range is from 70% to 100%) suggesting that model comparisons among models using a single run results can be unreliable. The accuracy distribution further indicates the competitive performance of Phi-4-reasoning-plus, largely intersecting with o3-mini's distribution and being almost disjoint from the DeepSeek-R1-Distill-Llama-70B's distribution.

baseline models were recently reported in [10], we reuse the same evaluation logs for baseline comparisons as well as the same evaluation methodology for studying Phi-4-reasoning and Phi-4-reasoning-plus.

The full list of reasoning benchmarks used in this report is summarized in Table 4. The performance of Phi-4-reasoning and Phi-4-reasoning-plus on this comprehensive suite of benchmarks along with select baselines are shown in Figure 8.

5.1.1 Baseline models

We compare our models with selected state-of-the-art models as described in Table 3, including o1, o3-mini-high, DeepSeek-R1, DeepSeek-R1-Distill-Llama-70B, Claude 3.7 Sonnet, and Gemini 2 Flash Thinking. We use temperature 0.8 for the Phi models, 0.6 for models in the DeepSeek family as recommended in the corresponding model cards on HuggingFace, and otherwise use either 1.0 or the default temperature setting in the model APIs. In terms of maximum token length, we aim to allow as many tokens as possible to all models, within limits that do not present other experimental hurdles such as timeouts from the API. For both our models we use the system message described in Section 3. We use the same CoT prompt templates for all models on all benchmarks, which explicitly require models to think step by step and then provide an answer. For o1 evaluations we use a plain non-CoT prompt template because of policy violation refusals usually triggered by CoT requests.

In addition to the baselines in Figure 8, we report comparison with evaluations in newer baselines including OpenThinker2 [53], QwQ [54], EXAONE-Deep [31], and Gemini-2.5-Pro [18] in Table 1.

5.1.2 Accuracy distribution on AIME 2025: Beyond single-score analyses

Most existing comparisons of reasoning models on AIME 2025 use the average accuracy computed on a single pass over the dataset. However, LLMs have exhibited large generation nondeterminism, *i.e.,* they may produce substantially different answers given the same prompts and inference parameters (such as temperature and max tokens) [9, 10, 25]. For larger non-reasoning models, nondeterminism can occur even at very low temperature (even zero), and the phenomenon is also almost always expected for reasoning models which are expected to diversify

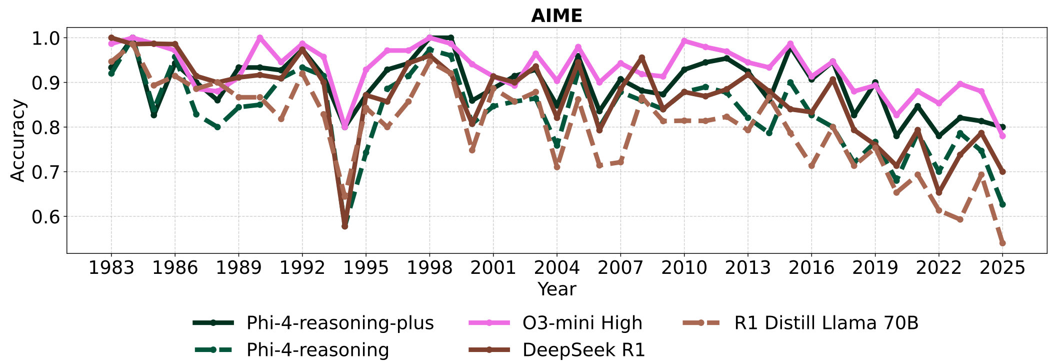


Figure 10: Performance breakdown by years (from 1983 to 2025) for AIME on 5 independent runs. There is a large performance variance across different years. For example, most models perform substantially worse in 1994 and 2025.

the inference paths and also highly recommended to be run at high temperatures (0.6 ~ 1.0). Given that AIME 2025 also only contains 30 questions, nondeterminism renders the accuracy-based analysis questionable.

To account for the stochastic nature of such experiments, we study the accuracy distribution on AIME 2025, approximated by kernel density estimation using 50 independent runs with the same prompt and temperature (see Table 3 for temperature details). We have found several interesting observations as shown in Figure 9. First, we observe a high accuracy variance for all models. For example, accuracy of answers generated by DeepSeek-R1-Distill-Llama-70B ranges from 30% to 70%, while o3-mini's accuracy range is from 70% to 100%. This suggests that any comparison among models using a single run can easily produce misleading conclusions. Second, models on the two extremes of average accuracy demonstrate more robust accuracy. For example, Phi-4-reasoning-plus and Phi-4 have relatively narrower accuracy ranges compared to DeepSeek-R1-Distill-Llama-70B and Phi-4-reasoning. Third, the accuracy distribution further indicates the competitive performance of Phi-4-reasoning-plus, largely intersecting with o3-mini's distribution and being almost disjoint from DeepSeek-R1-Distill-Llama-70B's distribution. Figure 8 also shows the average pass@1 accuracy for several models including ours along with the standard deviation. Note that given the very small size of the data, this picture can look dramatically different from experiments with fewer runs (e.g., 5) because there may exist variance even between two different sets of 5 independent runs. This behavior also explains differences and discrepancies between different and concurrent evaluation works that may report different scores across five runs [25, 46] which deviate from what model cards report at release time.

We hope that these results will serve as insights for more robust quantitative analyses and as a motivation for moving beyond single-score and single-run accuracy reportings. Throughout this report, we also aimed to scale up our quantitative analysis across a diversified set of benchmarks that have a larger number of samples between 800-4500, except AIME 24 & 25, HMMT and GPQA which are commonly present in most technical reports of recently released models. Larger benchmarks also enable more disaggregated error analysis that allows for analyzing model performance across different subgroups of data, finding common error patterns, and potentially behavioral correlations between models [42, 12, 40].

At the same time, it is still important to evaluate models on smaller but very challenging benchmarks (*e.g.,* HMMT, USAMO, AIME) if that analysis includes several runs, standard deviation and statistical tests, and preferably qualitative insights. For example, tools like MathArena [11] and our work on Eureka ML Insights [9] practice the process of revealing not only the scores but also complete evaluation logs per benchmark prompt and per run. In particular, for reasoning models, further work is needed to better understand the variance in model behavior and to study the properties of longer solution scratchpads.

5.1.3 Main findings

We present the main findings from the results on the broader set of reasoning benchmarks presented in Figures 8 and Table 1. We also present complementary analysis that support additional discussions in this section.

Reasoning benchmark performances. Phi-4-reasoning and Phi-4-reasoning-plus both present major improvements over the earlier Phi-4 model on a wide range of reasoning tasks including math and scientific reasoning, coding, algorithmic problem solving, and planning. In particular, the two new models improve by 50% accuracy on math benchmarks (AIME and Omni-Math) and by over 25% on coding (LiveCodeBench). Surprisingly, these models also improve by 30%-60% on algorithmic and planning problems (Traveling Salesman, Satisfiability, Calendar Planning) which demonstrates increased generalizability on domains that were not targeted in the fine-tuning or RL training for reasoning.

Phi-4-reasoning and Phi-4-reasoning-plus are comparable with or have better accuracy than R1 models (DeepSeek-R1 and DeepSeek-R1-Distill-Llama-70B) and o1/o3-mini models on math reasoning, despite being smaller models with only 14B parameters. While o1/o3-mini have very strong performance on AIME which is a popularly reported and small-sized benchmark, the efficacy reduces when considering a more diverse benchmark like Omni-MATH as they struggle in domains like Discrete Math and Geometry (see Figure 15). Phi-4-reasoning and Phi-4-reasoning-plus show strong generalization to these diverse math settings as well. This showcases the benefits of rich data synthesis processes based on high-quality seeds and scalable distillation processes, further supported by enhanced exploration at RL stage. They also outperform Claude 3.7 Sonnet and Gemini 2 Flash Thinking on all tasks except GPQA and Calendar Planning.

Phi-4-reasoning-plus marks important advantages over Phi-4-reasoning on math (also targeted during the RL stage). More specifically, Phi-4-reasoning-plus is 15% more accurate than Phi-4-reasoning on average across 50 runs on AIME 2025 (Figure 8), and 5% more accurate on Omni-Math and TSP. This is less pronounced on algorithmic reasoning, planning, coding, and spatial understanding tasks, encouraging further work that enables exploration and verification for a broader set of tasks. For example, we observe that even though the generations of Phi-4-reasoning-plus are on average across tasks 1.5x longer, on tasks like Calendar Planning, 3SAT, Maze, and SpatialMap, the lengthened traces do not lead to higher accuracy. Nevertheless, a deeper analysis by difficulty level presented in Figure 14 shows that Phi-4-reasoning-plus is still able to offer improvements for the easy-to-mid levels of difficulty. Accuracy drops towards higher difficulty levels are common for all models, even the most capable ones.

To evaluate the models coding ability, we report LiveCodeBench (2024-08 – 2025-01) and Codeforces (using the `143` problems from contest IDs 1505 through 1536 inclusive) benchmarks in Table 1. For Codeforces, we allow each model ten independent submission attempts for every problem and finally compute the Elo rating. To validate comparison against other models, we evaluate the same Codeforces protocol on DeepSeek-R1-Distil-Qwen-14B and receive Elo score of `1481`. Our numbers were within 1 accuracy point for LiveCodeBench and within 20 Elo rating points for CodeForces as reported in [21]. For all other models, we report the Codeforces numbers as published in their corresponding reports for reference, however, they may not be directly comparable due to the lack of public information on their exact evaluation procedure.

The evaluation also reveals several opportunities for improvement for both models we contribute in this report, and for other models evaluated alongside. As previously noted in recent work [10], improvements in scientific topics like biology and chemistry are smaller than for math and physics (Figure 16). Even within math, all models have lower accuracy on sub topics like discrete math, indicating areas for improvement (Figure 15). Interestingly, models also follow similar, potentially correlated patterns of errors across years in AIME competitions between 1983-2025 (Figure 10). Most prominently, all models' performance drops over time and for recent years.

Performance vs. token usage tradeoffs. Results on accuracy vs. token usage tradeoffs are shown in Figure 11. In terms of token length, on average over reasoning benchmarks Phi-4-reasoning-plus's generations are 1.5x longer than Phi-4-reasoning, and 1.15x longer than o3-mini. Phi-4-reasoning token lengths are similar to o3-mini. The horizontal error bars in the chart show the standard deviation per instance (prompt), which means that standard deviation was first computed for each prompt, and then averaged across all prompts in the benchmark. This is to show generation length variability for the same prompt, which is comparable for all models.

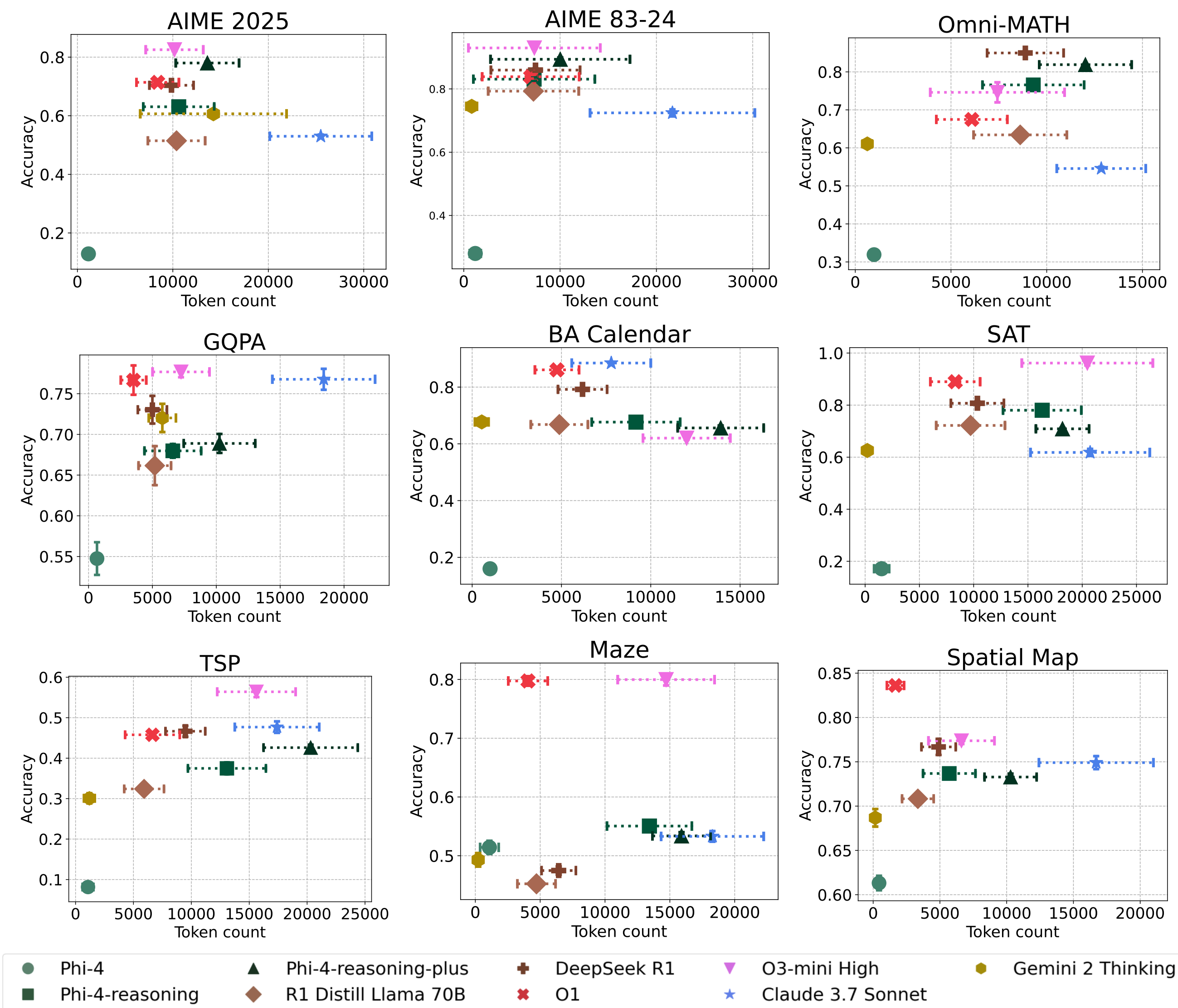


Figure 11: Tradeoff between accuracy and token usage for all benchmarks. The standard deviation for accuracy (vertical, filled line) is computed across 5 different repetitions (50 for AIME 2025). The standard deviation for token usage (horizontal, dotted line) is computed by first taking the standard deviation per data instance, and then averaging by the size of the benchmark, to show the variability per instance.

5.1.4 Average vs. best-of-5 performance

Figures 12 and 17 show different aggregation scores for five runs on all benchmarks. Comparisons between average accuracy scores across 5 runs and best-of-5 scores show that, similarly to all other reasoning models, there may exist even better trajectories in our models' generations revealing potential for further progress. This also affirms the importance of using improved verifiers at training time that can extract model capabilities which we cannot yet access. While these insights are encouraging, extracting such capabilities without spending N-times more inference compute on models that are already expensive at inference time, remains an open question for future work. Reliably improving accuracy across many samplings from the same prompt remains a caveat for all models,

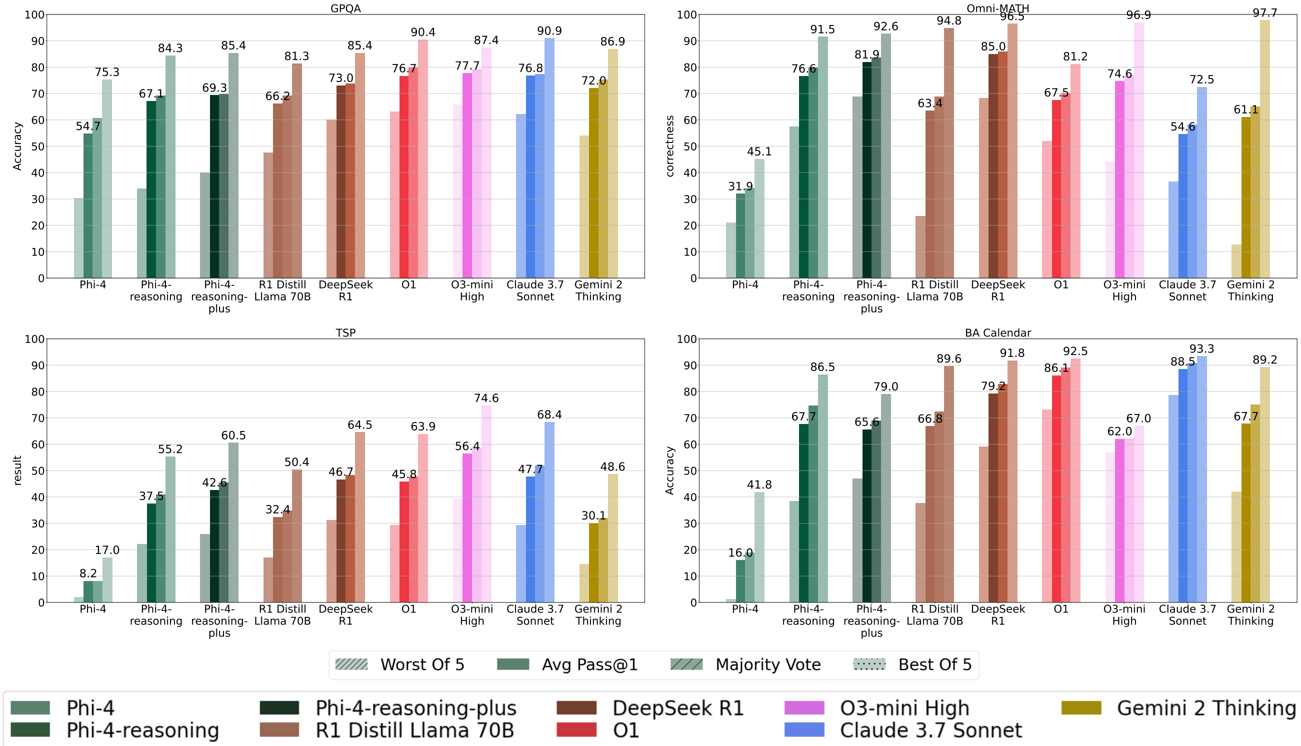


Figure 12: Results on reasoning benchmarks with different aggregations on 5 independent runs: worst of 5, average pass@1, majority vote, and best of 5. An extended version of this figure can be found in appendix Figure 17.

including ours. This is an aspect that can impact user and developer experience on repeated prompts, where determinism and predictable performance are both important.

5.2 General-purpose Benchmarks

In addition to reasoning benchmarks, we also report results in standard benchmarks to ensure training models on reasoning does not degrade their general abilities. First, we use the benchmarks from the Phi-4 report [2]. Specifically, we use OpenAI's simple-evals framework (including prompts, temperature, and extraction) for evaluating the model on MMLU [24] and MGSM [50]. Furthermore, we evaluate our models on MMLU-pro [57], HumanEvalPlus [35], and ArenaHard [33]. Next, we also use a set of general-purpose benchmarks from the Eureka ML Insights repository, namely FlenQA [30], Toxigen [23], Kitab [3] and IFEval [62][7]. Finally, we use PhiBench, our internal collection of evaluations [2]. See Table 2 for results.

FlenQA [30] consists of 12K questions ranging in length from 250 to 3000 tokens with True/False labels. This benchmark was designed to isolate the effect of input length on LLMs' performance using multiple versions of the same task, extended with padding of different lengths, types and locations. The task itself involves making inferences on top of two logical statement needles introduced in context, and answering a question that requires both statements. Each prompt is padded with paragraphs sampled from other instances of the same task, or paragraphs sampled from Book Corpus, with key information presented at various locations in the context (at the beginning, at the end, in the middle, or at random locations). Results on this benchmark show that reasoning models, Phi-4-reasoning, Phi-4-reasoning-plus, and o3-mini, are robust to longer inputs compared to conventional models Phi-4 and GPT-4o, and they are also not affected by the location of key information in the context (See

[7] Evaluations of Phi-4 on Eureka ML Insights were done at temperature=0 and reported for a single run all benchmarks except IFEval which is smaller than the other benchmarks, for which we ran the evaluation three times (for all models) and report the average accuracy.

Model	Phi-4	Phi-4-reasoning	Phi-4-reasoning-plus	o3-mini	GPT-4o
FlenQA [3K-token subset] [30]	82.0	97.7	**97.9**	96.8	90.8
IFEval Strict [62]	62.3	83.4	84.9	**91.5**	81.8
ArenaHard [33]	68.1⋆	73.3	79.0	**81.9**	69.0⋆
HumanEvalPlus [35]	83.5	92.9	92.3	**94.0**	84.9
MMLUPro [57]	71.5	74.3	76.0	**79.4**	73.5
Kitab [3]					
No Context - Precision	19.3	23.2	27.6	37.9	**53.7**
With Context - Precision	88.5	93.8	93.6	**94.0**	84.7
No Context - Recall	8.2	4.9	6.3	4.2	**20.3**
With Context - Recall	68.1	74.8	75.4	**76.1**	69.2
Toxigen Discriminative [23]					
Toxic category	72.6	86.7	77.3	85.4	**87.6**
Neutral category	90.0	84.7	**90.5**	88.7	85.1
PhiBench 2.21 [2]	58.2	70.6	74.2	**78.0**	73.1

Table 2: Average pass@1 accuracy of models across general-purpose benchmarks evaluated averaged over five generations. For the results in this table Phi-4-reasoning and Phi-4-reasoning-plus were evaluated at temperature 0.8, while Phi-4 was evaluated at temperature of 0.0. For ArenaHard⋆, there is a discrepancy for Phi-4 and GPT-4o compared to the numbers reported in the Phi-4 paper [2] due to changing the backend LLM-judge. Bold and underlined numbers denote the best and second best scores, respectively, per benchmark.

appendix Figure 13). Improvements are potentially linked to the fact that reasoning models are trained to better handle longer context and self-reflection on such context.

Kitab [3] is a challenging information retrieval benchmark containing queries with constraint filters (e.g. List all books written by Isabel Allende, written between 2000-2010.). We performed Kitab evaluations either using the model's parametric knowledge only (no context) or with grounding in a RAG-style setting (with context), only on the subset of queries that contains a single book constraint. The addition of reasoning capabilities to the Phi family models generally seems to improve precision and degrade recall in the no-context setting, while improving both metrics to be almost on-par with o3-mini when retrieval context is provided. Information retrieval and factuality with only parametric knowledge remain challenging for our models, potentially due to their significantly smaller scale. This remains an avenue for further improvement for all models (even for the larger ones), for teaching them how and when to retrieve information from other sources.

IFEval [62] includes instruction-based prompts that involve instructions amenable to objective verification of compliance. The addition of reasoning capabilities to the Phi family significantly improves performance on this benchmark to even surpass GPT-4o's performance.

Our reasoning models also lead to improvements over the Phi-4 model of more than 10% on ArenaHard, HumanEvalPlus, and our private PhiBench 2.21 data, which is tailored to evaluate diverse abilities that the team found critical to Phi-4 development [2]. There is also a 3%-5% improvement on MMLUPro. As we mentioned earlier in this section, while these benchmarks are more for general purpose evaluation, it is not entirely incidental that we observe improvements on these too. These benchmarks still combine simpler forms of reasoning as part of the main task, although reasoning is not the main skill to test. Some of them even include simpler math problems and constraint satisfaction queries. It is however encouraging to see that Phi-4-reasoning and Phi-4-reasoning-plus bring well-rounded, general improvements on a highly diverse set of measurements.

5.3 Safety Evaluation

We developed Phi-4-reasoning in accordance with Microsoft's Responsible AI principles[8]. To assess model safety, we used the following benchmarks:

[8]https://www.microsoft.com/en-us/ai/principles-and-approach

Figure 13: FlenQA - (Left) Effect of context length on language models' performance. All models suffer a degradation in accuracy with increased context length. Reasoning models are more robust than conventional models in this regard. (Right) Accuracy of language models in the longest context setting (3000 tokens), with key information placed at the beginning (*first*), at the end (*last*), in the middle (*middle*), or dispersed at random locations (*random*) of the context. In the first three dispersion settings, the two pieces of key information are adjacent, while in the *random* setting they are presented separately. Conventional models are remarkably more challenged in the *random* setting compared to other settings while reasoning models are not affected by the dispersion of information.

Automated RAI Measurement Framework: We used a framework for automated measurement of Responsible AI metrics for LLMs [37]. In this framework, one LLM poses as a user and engages in a synthetic conversation with the LLM under test. Another LLM then evaluates the responses to measure its tendency to violate Responsible AI guidelines for the following categories: (1) How many times the user succeeded in Jailbreaking the model under test? (2) How many times the model under test generates Potentially Harmful Content? (3) How many times the model leaks Intellectual Property (IP)? These categories are divided into multiple sub-categories described in [37]. Results show that Phi-4-reasoning shows minor regression compared to the base model Phi-4.

Toxigen: Toxigen [23] is a large-scale dataset consisting of toxic and neutral statements about 13 groups of people with a focus on implicit hate speech about minority groups. The dataset is balanced with an equal number of toxic and neutral prompts for each identity group. This allows us to evaluate erasure as well as toxicity detection: that is where the identity mention of specific groups of people is treated as a signal of toxicity or hate. Results indicate that as we add reasoning capabilities to models, we are able to improve performance on only one of the toxic or neutral categories at a time, indicating that detecting toxicity without causing erasure is still a challenging problem for all models. Nevertheless, we still observe modest improvements even in this task, with Phi-4-reasoning offering a better balance (than Phi-4-reasoning-plus and Phi-4) between toxic vs. neutral content detection accuracy. Lower imbalance is also a preferable behavior for applications such as content filtering and moderation. Figure 18b in the appendix shows a more detailed view of model accuracy on Toxigen, reported for 13 different demographic groups. Improvements from Phi-4-reasoning show that the model has also narrowed some of the group-based discrepancies, previously observed in Phi-4. This is a positive direction towards increased group-based fairness. Both models also improve upon the Phi-4 model.

Despite the above measurements, appropriate evaluation of output generated by reasoning models remains a challenge mainly due to the fact that current LLM judges and tools have not yet been optimized for long, step-by-step traces and may struggle with the extended length and the non-linear generations. For example, reasoning models may tend to repeat the problem, create counterfactuals of the problem statement or edge cases. In this process, judges may falsely trigger safety measures or otherwise miss important assumptions and biases that models might have due to the entangled nature of the language [29]. Future research is needed to mature the practices in these areas by potentially improving safety-oriented verification methods and by decomposing and simplifying the evaluation of long traces.

Finally, the above measurements focus on measuring engagement and refusal in jailbreak scenarios and on classification-style toxic language detection. We acknowledge that it is also important to evaluate models, including ours, in more open-form and benign scenarios, beyond jailbreaks and adversarial scenarios. These evaluations would enable a better understanding of how biases and harms may occur in benign, real-world tasks.

6 Limitations

Phi-4-reasoning inherits limitations from its base model. For example, the Phi-4 model primarily supports English text, with performance declining for other languages and less represented English varieties compared to standard American English. Despite safety measures, the model may still perpetuate stereotypes, misrepresent groups, or produce inappropriate content due to biases in training data. Additionally, as we show in our evaluations on general-purpose benchmarks for factuality, the model can generate inaccurate or outdated information that sounds plausible. For coding, Phi-4 is mainly trained on Python using common packages, and users should manually verify API uses if scripts involve other languages or packages.

Additionally, Phi-4-reasoning also exhibits common limitations of other reasoning language models such as requiring more computational time and resources due to their reasoning processes, resulting in slower response times compared conventional LLMs. Additionally, reasoning models may produce responses that contradict their own reasoning chains, potentially leading to inaccuracies. The ability to understand and monitor the reasoning steps for more transparency is still an active area of research.

The Phi-4-reasoning model, while powerful, also has notable limitations, particularly with its context length of 32k tokens. This constraint can be limiting for more complex tasks that require extensive context to generate accurate and coherent responses. Additionally, the model sometimes generates responses that exceed its context window size, leading to truncation and loss of important information. This limited context window size also impacts the model's performance in multiturn interactions, as it may struggle to maintain continuity over extended conversations. Furthermore, the supervised fine-tuning (SFT) training data is limited to STEM, code, and safety, while the reinforcement learning (RL) data is limited to math. Although there are signs of generalization to other domains, this limitation may affect the model's performance on different contexts.

Author Contributions

Data and Supervised Finetuning: Mojan Javaheripi, Arindam Mitra, Sahaj Agarwal, Caio César Teodoro Mendes, Olli Saarikivi, Marah Abdin and Suriya Gunasekar
Reinforcement Learning: Yue Wu, Harkirat Behl, Guoqing Zheng, Vaish Shrivastava and Dimitirs Papailiopoulos
Evaluation and Analysis: Vidhisha Balachandran, Lingjiao Chen, Neel Joshi, Vibhav Vineet, Safoora Yousefi and Besmira Nushi
Infrastructure and Release: Yash Lara, Gustavo de Rosa, Piero Kauffmann and Shital Shah
Project Lead: Ahmed Awadallah

Acknowledgements

We would like to thank Eric Price, Shivam Garg, John Langford for discussion and feedback and Ece Kamar, Rafah Hosn, Doug Burger, Peter Lee, Asha Sharma, Chelsea Carter, Rob Young and Weishung Liu for their support throughout the project.

References

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[2] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Xin Wang, Rachel Ward, Yue Wu, Dingli Yu, Cyril Zhang, and Yi Zhang. Phi-4 technical report, 2024. URL `https://arxiv.org/abs/2412.08905`.

[3] Marah I Abdin, Suriya Gunasekar, Varun Chandrasekaran, Jerry Li, Mert Yüksekgönül, Rahee Ghosh Peshawaria, Ranjita Naik, and Besmira Nushi. KITAB: evaluating llms on constraint satisfaction for information retrieval. In *International Conference on Learning Representations*, 2024.

[4] AIME. Aime 83-24. `https://huggingface.co/datasets/lchen001/AIME1983_2024`, 2024. Accessed: 2025-03-17.

[5] AIME. Aime 2025. `https://huggingface.co/datasets/lchen001/AIME2025`, 2025. Accessed: 2025-03-17.

[6] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[7] Anthropic. Claude 3.7 sonnet. `https://www.anthropic.com/news/claude-3-7-sonnet`, 2025. Accessed: 2025-03-17.

[8] Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-thought reasoning in the wild is not always faithful. *arXiv preprint arXiv:2503.08679*, 2025.

[9] Vidhisha Balachandran, Jingya Chen, Neel Joshi, Besmira Nushi, Hamid Palangi, Eduardo Salinas, Vibhav Vineet, James Woffinden-Luey, and Safoora Yousefi. Eureka: Evaluating and understanding large foundation models. *arXiv preprint arXiv:2409.10566*, 2024.

[10] Vidhisha Balachandran, Jingya Chen, Lingjiao Chen, Shivam Garg, Neel Joshi, Yash Lara, John Langford, Besmira Nushi, Vibhav Vineet, Yue Wu, and Safoora Yousefi. Inference-time scaling for complex tasks: Where we stand and what lies ahead, 2025. URL `https://arxiv.org/abs/2504.00294`.

[11] Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions, February 2025. URL `https://matharena.ai/`.

[12] Solon Barocas, Anhong Guo, Ece Kamar, Jacquelyn Krones, Meredith Ringel Morris, Jennifer Wortman Vaughan, W Duncan Wadsworth, and Hanna Wallach. Designing disaggregated evaluations of ai systems: Choices, considerations, and tradeoffs. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 368–378, 2021.

[13] Natasha Butt, Varun Chandrasekaran, Neel Joshi, Besmira Nushi, and Vidhisha Balachandran. Benchagents: Automated benchmark creation with agent interaction. *arXiv preprint arXiv:2410.22584*, 2024.

[14] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023.

[15] Quy-Anh Dang and Chris Ngo. Reinforcement learning for reasoning in small llms: What works and what doesn't, 2025. URL `https://arxiv.org/abs/2503.16219`.

[16] Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, et al. Omni-math: A universal olympiad level mathematic benchmark for large language models. *ICLR*, 2025.

[17] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR, 2023.

[18] Google. Gemini flash thinking. `https://deepmind.google/technologies/gemini/flash/`, 2025. Accessed: 2025-03-17.

[19] Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *arXiv preprint arXiv:2501.04519*, 2025.

[20] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

[21] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[22] Juris Hartmanis. Computers and intractability: a guide to the theory of np-completeness (michael r. garey and david s. johnson). *Siam Review*, 24(1):90, 1982.

[23] Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326. Association for Computational Linguistics, 2022.

[24] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL `https://arxiv.org/abs/2009.03300`.

[25] Andreas Hochlehnert, Hardik Bhatnagar, Vishaal Udandarao, Samuel Albanie, Ameya Prabhu, and Matthias Bethge. A sober look at progress in language model reasoning: Pitfalls and paths to reproducibility. *arXiv preprint arXiv:2504.07086*, 2025.

[26] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[27] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[28] Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, Suriya Gunasekar, Piero Kauffmann, Yin Tat Lee, Yuanzhi Li, Anh Nguyen, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Michael Santacroce, Harkirat Singh Behl, Adam Taumann Kalai, Xin Wang, Rachel Ward, Philipp Witte, Cyril Zhang, and Yi Zhang. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 2023.

[29] Fengqing Jiang, Zhangchen Xu, Yuetai Li, Luyao Niu, Zhen Xiang, Bo Li, Bill Yuchen Lin, and Radha Poovendran. Safechain: Safety of language models with long chain-of-thought reasoning capabilities. *arXiv preprint arXiv:2502.12025*, 2025.

[30] Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *ACL*, 2024.

[31] LG AI Research. Exaone deep: Reasoning enhanced language models. *arXiv preprint arXiv:2503.12524*, 2025.

[32] Shanda Li, Chong You, Guru Guruganesh, Joshua Ainslie, Santiago Ontanon, Manzil Zaheer, Sumit Sanghai, Yiming Yang, Sanjiv Kumar, and Srinadh Bhojanapalli. Functional interpolation for relative positions improves long context transformers. *arXiv preprint arXiv:2310.04418*, 2023.

[33] Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*, 2024.

[34] Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling, 2025.

[35] Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation, 2023. URL `https://arxiv.org/abs/2305.01210`.

[36] Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. https://pretty-radio-b75.notion.site/DeepScaleR-Surpassing-O1-Preview-with-a-1-5B-Model-by-Scaling-RL-19681902c1468005bed8ca303013a4e2, 2025. Notion Blog.

[37] Ahmed Magooda, Alec Helyar, Kyle Jackson, David Sullivan, Chad Atalla, Emily Sheng, Dan Vann, Richard Edgar, Hamid Palangi, Roman Lutz, Hongliang Kong, Vincent Yun, Eslam Kamal, Federico Zarfati, Hanna Wallach, Sarah Bird, and Mei Chen. A framework for automated measurement of responsible ai harms in generative ai applications, 2023. URL https://arxiv.org/abs/2310.17750.

[38] Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*, 2023.

[39] Arindam Mitra, Luciano Del Corro, Guoqing Zheng, Shweti Mahajan, Dany Rouhana, Andres Codas, Yadong Lu, Wei-ge Chen, Olga Vrousgos, Corby Rosset, et al. Agentinstruct: Toward generative teaching with agentic flows. *arXiv preprint arXiv:2407.03502*, 2024.

[40] Mazda Moayeri, Vidhisha Balachandran, Varun Chandrasekaran, Safoora Yousefi, Thomas Fel, Soheil Feizi, Besmira Nushi, Neel Joshi, and Vibhav Vineet. Unearthing skill-level insights for understanding trade-offs of foundation models. *ICLR*, 2025.

[41] Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*, 2023.

[42] Besmira Nushi, Ece Kamar, and Eric Horvitz. Towards accountable ai: Hybrid human-machine analyses for characterizing system failure. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 6, pages 126–135, 2018.

[43] OpenAI. Openai o3-mini system card. https://openai.com/index/o3-mini-system-card/, 2025. Accessed: 2025-03-17.

[44] Christos H Papadimitriou. Computational complexity. In *Encyclopedia of computer science*, pages 260–265. John Wiley and Sons Ltd., 2003.

[45] Samir Passi and Mihaela Vorvoreanu. Overreliance on ai literature review. *Microsoft Research*, 339:340, 2022.

[46] Ivo Petrov, Jasper Dekoninck, Lyuben Baltadzhiev, Maria Drencheva, Kristian Minchev, Mislav Balunović, Nikola Jovanović, and Martin Vechev. Proof or bluff? evaluating llms on 2025 usa math olympiad. *arXiv preprint arXiv:2503.21934*, 2025.

[47] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

[48] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[49] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

[50] Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language models are multilingual chain-of-thought reasoners, 2022. URL https://arxiv.org/abs/2210.03057.

[51] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

[52] Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

[53] OpenThoughts Team. Open Thoughts. https://open-thoughts.ai, January 2025.

[54] Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, March 2025. URL https://qwenlm.github.io/blog/qwq-32b/.

[55] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting, 2023. URL https://arxiv.org/abs/2305.04388.

[56] Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Advances in Neural Information Processing Systems*, 37:75392–75421, 2024.

[57] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark, 2024. URL https://arxiv.org/abs/2406.01574.

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

[59] Guanghao Ye, Khiem Duc Pham, Xinzhi Zhang, Sivakanth Gopi, Baolin Peng, Beibin Li, Janardhan Kulkarni, and Huseyin A Inan. On the emergence of thinking in llms i: Searching for the right intuition. *arXiv preprint arXiv:2502.06773*, 2025.

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

[61] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

[62] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Benchmarking Details

Model	temp.	max token	reasoning
Phi-4 [2]	0.8†	4,096	n
Phi-4-reasoning	0.8	32,768*	y
Phi-4-reasoning-plus	0.8	32,768*	y
DeepSeek-R1-Distill-Llama-70B [21]	0.6	65,536	y
DeepSeek-R1 [21]	0.6	65,536	y
GPT-4o 2024-05-13 [26]	1.0	4,096	n
o1-mini 2024-09-12 [27]	NA	NA	y
o1 2024-12-17 [27]	NA	NA	y
o3-mini 2025-01-31(high) [43]	NA	NA	y
Claude 3.7 Sonnet 2025-02-19 [7]	1.0	32,768	y
Gemini 2 Flash Thinking Exp 2025-01-21 [18]	1.0	32,768	y

Table 3: List of models studied in this report and corresponding temperature and maximum token limits used for all experiments. *For Phi-4-reasoning and Phi-4-reasoning-plus evaluations on AIME, HMMT, GPQA, and Codeforces we use 65,536 as the maximum number of tokens for generation without changing any RoPE parameters. We note that neither model has properly trained on this length. All other evaluations for Phi-4-reasoning and Phi-4-reasoning-plus use 32,768. †For Phi-4 we use temp=0.8 for the reasoning benchmarks, and 0.0 for the general-purpose benchmarks.

Dataset (#prompts)	Link
AIME 25 [5] (30)	`https://huggingface.co/datasets/lchen001/AIME2025`
AIME 83-24 [4] (949)	`https://huggingface.co/datasets/lchen001/AIME1983_2024`
HMMT February 2025 (30)	`https://huggingface.co/datasets/MathArena/hmmt_feb_2025`
Omni-MATH [16] (4428)	`https://huggingface.co/datasets/KbsdJames/Omni-MATH`
GPQA◇ [47] (198)	`https://huggingface.co/datasets/Idavidrein/gpqa`
BA-Calendar [13] (2000)	`https://huggingface.co/datasets/microsoft/ba-calendar`
TSP-Opt (new benchmark) (960)	To be released
3SAT-Search (new benchmark) (800)	To be released
Maze [56] (1500)	`https://huggingface.co/datasets/microsoft/VISION_LANGUAGE`
SpatialMap [56] (1500)	`https://huggingface.co/datasets/microsoft/VISION_LANGUAGE`

Table 4: List of reasoning benchmarks used in this report and where to find them.

B Additional Results

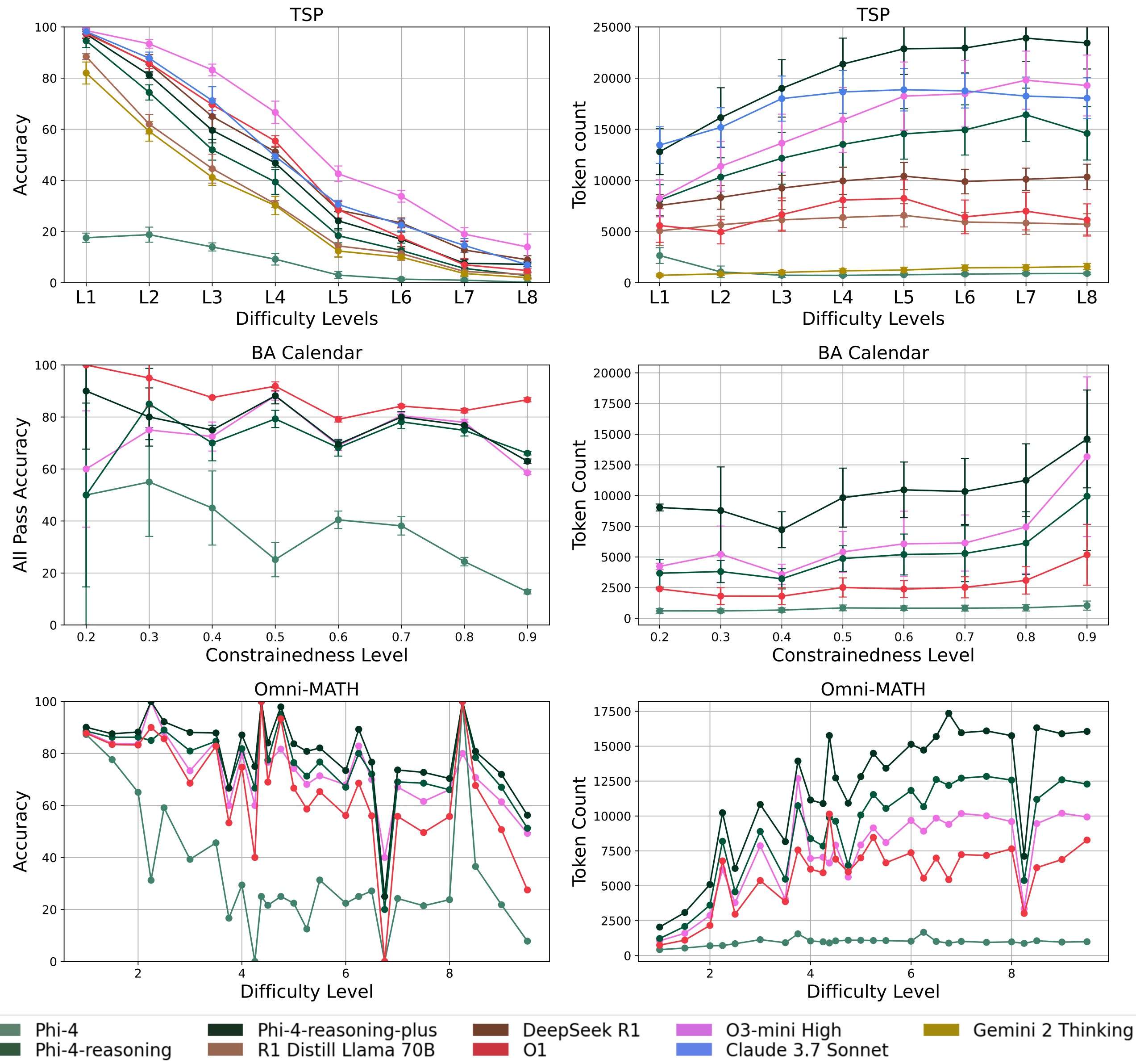


Figure 14: TSP, BA-Calendar, and Omni-Math accuracy and token usage with difficulty levels. Standard deviation for token usage is computed across different parallel repeats. For Omn-Math, we exclude error bars and some of the models to improve readability. Note the difficult tags in Omni-MATH can be nosiy and imperfect, and the size of the data available per level is imbalanced. This leads to fluctuations between adjacent difficulty levels. Similarly, we do not show all models for the BA-Calendar charts.

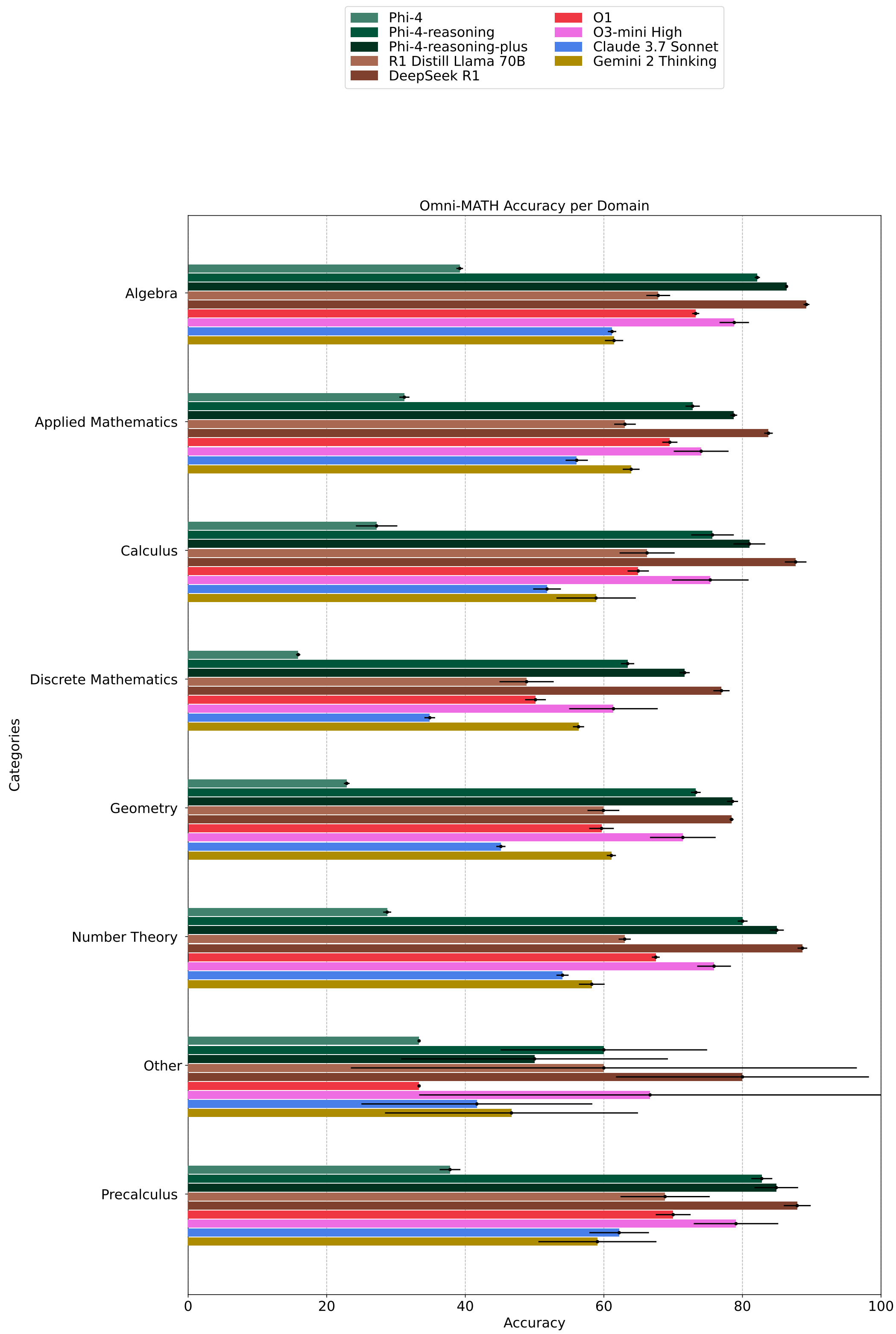


Figure 15: Omni-MATH topic-level accuracy. While general model trends are consistent across different topics, all models display lower performance on problems related to discrete mathematics and geometry.

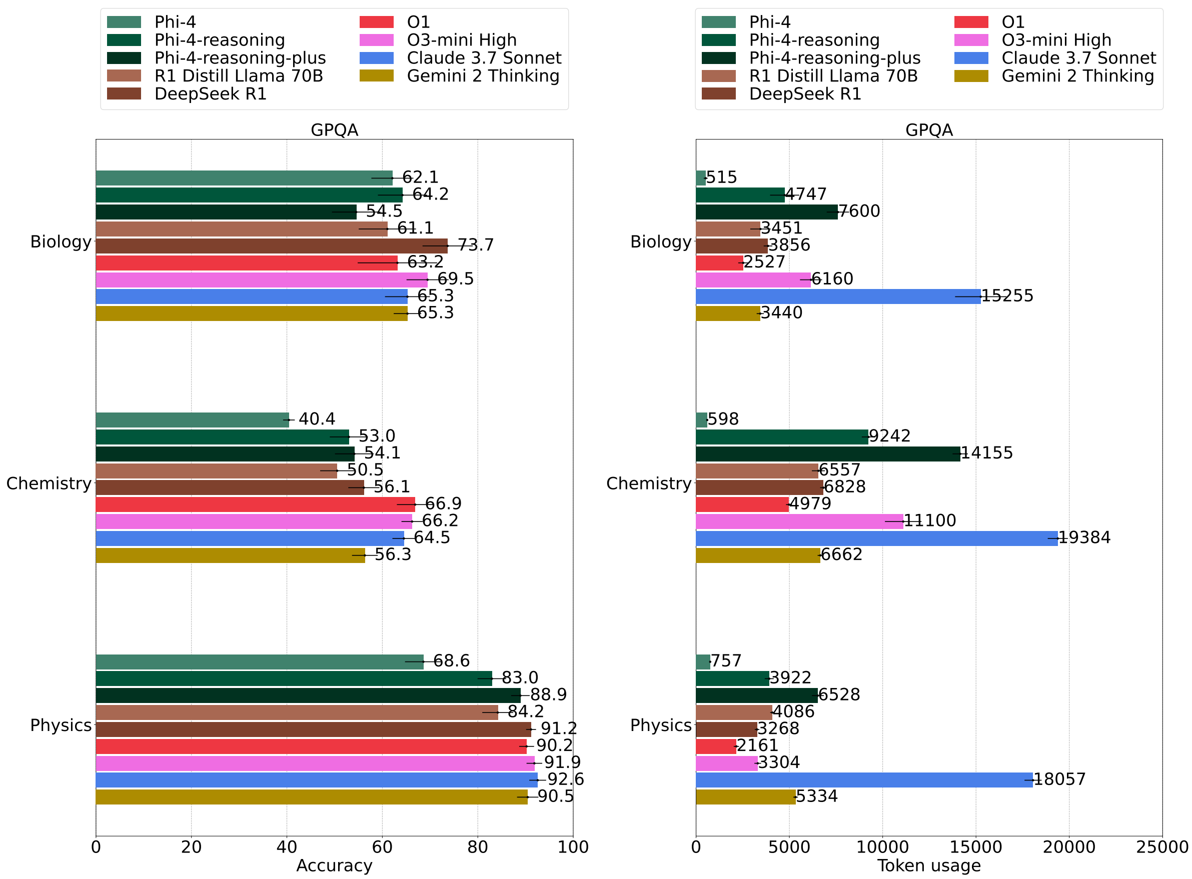


Figure 16: GPQA accuracy and token usage by high-level domain. Standard deviations for token usage are computed across five repeats, within the same high-level domain. Improvements for all models on biology and chemistry are lower than on physics.

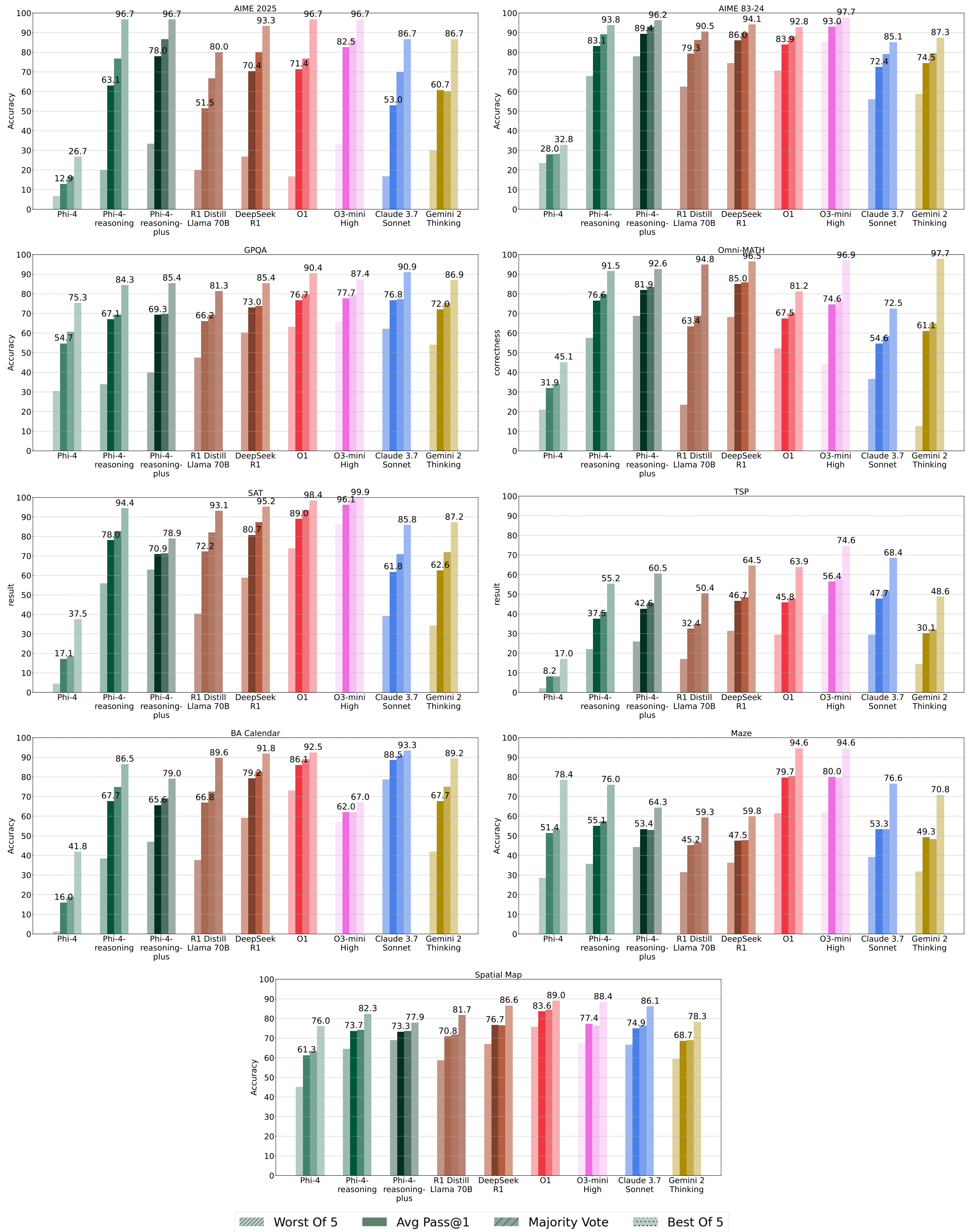
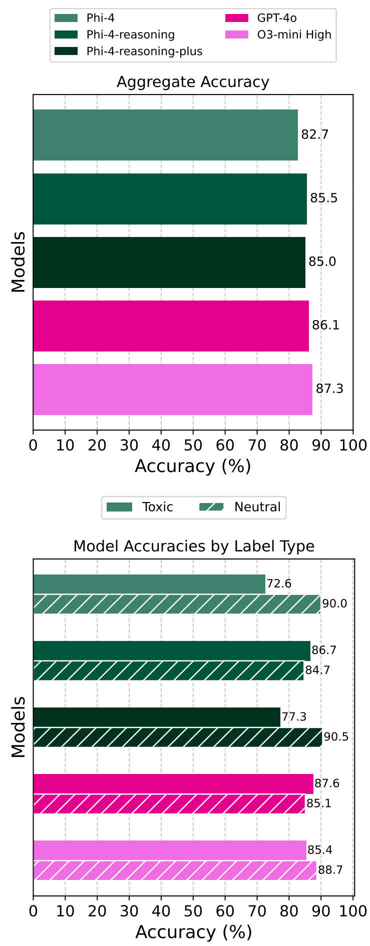
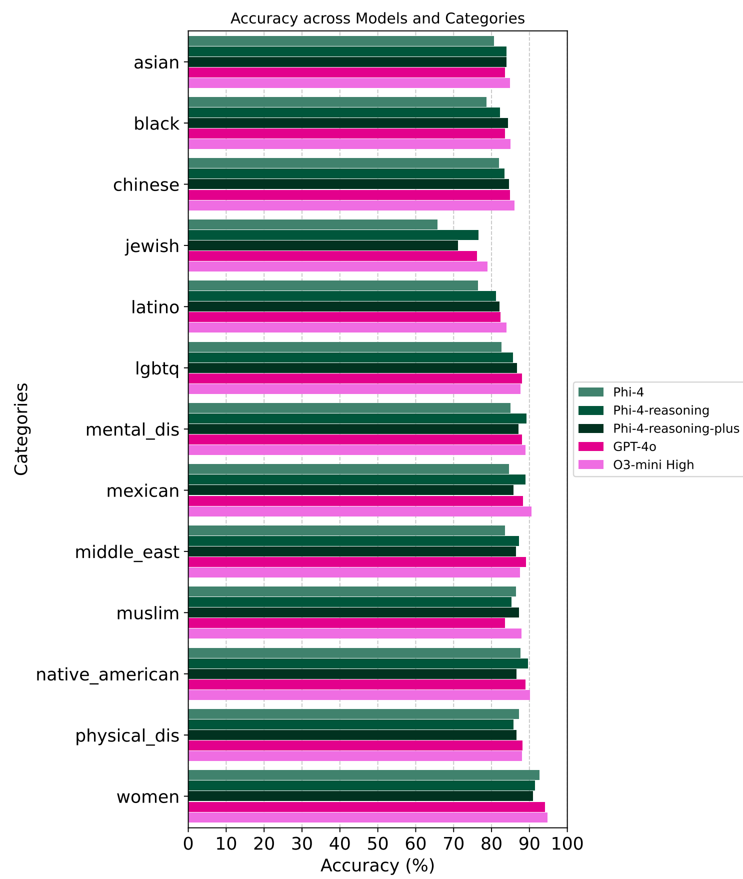


Figure 17: Results on reasoning benchmarks with different aggregations on 5 independent runs: worst of 5, average pass@1, majority vote, and best of 5. Across models and benchmarks we observe improvement in performance with best-of-5 indicating further room for improvement.



(a) Toxigen discriminative evaluation across different models.



(b) Model comparison across different categories in the discriminative evaluation setting of Toxigen.

Figure 18: Fine grained results on Toxigen. We observe modest improvements in this task of detecting toxic and neutral text, with Phi-4-reasoning offering a better balance between toxic vs. neutral content detection accuracy. Improvements from Phi-4-reasoning has also narrowed some of the group-based discrepancies, previously observed in Phi-4 indicating increased group-based fairness.