OPENMATHINSTRUCT-2: ACCELERATING AI FOR MATH WITH MASSIVE OPEN-SOURCE INSTRUCTION DATA

Shubham Toshniwal[1]   **Wei Du**[1]   **Ivan Moshkov**[1]
Branislav Kisacanin[1,2,3]   **Alexan Ayrapetyan**[1]   **Igor Gitman**[1]
`{stoshniwal,wedu,imoshkov,bkisacanin,aayrapetyan,igitman}@nvidia.com`

[1]NVIDIA
[2]Institute for AI R&D of Serbia
[3]Faculty of Technical Sciences, University of Novi Sad

ABSTRACT

Mathematical reasoning continues to be a critical challenge in large language model (LLM) development with significant interest. However, most of the cutting-edge progress in mathematical reasoning with LLMs has become *closed-source* due to lack of access to training data. This lack of data access limits researchers from understanding the impact of different choices for synthesizing and utilizing the data. With the goal of creating a high-quality finetuning (SFT) dataset for math reasoning, we conduct careful ablation experiments on data synthesis using the recently released `Llama3.1` family of models. Our experiments show that: (a) solution format matters, with excessively verbose solutions proving detrimental to SFT performance, (b) data generated by a strong teacher outperforms equally-sized data generated by a weak student model, (c) SFT is robust to low-quality solutions, allowing for imprecise data filtering, and (d) question diversity is crucial for achieving data scaling gains. Based on these insights, we create the OpenMathInstruct-2 dataset which consists of 14M question-solution pairs ($\approx$ 600K unique questions), making it nearly eight times larger than the previous largest open-source math reasoning dataset. Finetuning the `Llama-3.1-8B-Base` using OpenMathInstruct-2 outperforms `Llama3.1-8B-Instruct` on MATH by an absolute 15.9% (51.9% $\rightarrow$ 67.8%). Finally, to accelerate the open-source efforts, we release the code, the finetuned models, and the OpenMathInstruct-2 dataset under a commercially permissive license.[1]

1 INTRODUCTION

Synthetic data has emerged as a key technique for building large language models due to its cost-effectiveness and scalability (Meta-AI, 2024; NVIDIA, 2024; DeepSeek-AI, 2024). In particular, synthetic data is well suited for mathematical reasoning where the performance improvements with synthetic data scaling are yet to saturate (Zeng et al., 2024; Chan et al., 2024; Yang et al., 2024). However, access to this progress is limited because the current largest math datasets remain *closed-source* (Zeng et al., 2024; Yang et al., 2024). The closed nature of these datasets introduces two major issues. First, concerns over data leakage erode trust in reported benchmark results (Aiyappa et al., 2023). E.g., Zhang et al. (2024b) show a drop of more than 10% for popular LLMs on an unpublished test set which is distributionally similar to the popular grade school math benchmark GSM8K (Cobbe et al., 2021). Second, it prevents practitioners from fully understanding the impact of data composition and algorithmic choices (Azerbayev et al., 2024; Soldaini et al., 2024).

---

[1]Data and models are available at `https://huggingface.co/collections/nvidia/openmath-2-66fb142317d86400783d2c7b`
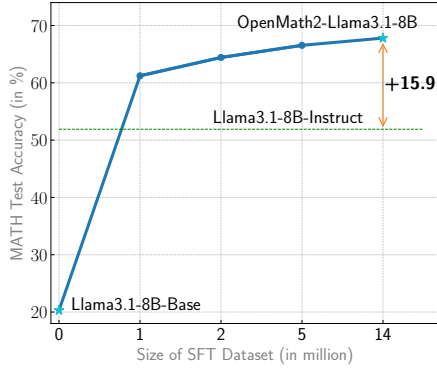Code is available at `https://github.com/NVIDIA/NeMo-Skills`

Figure 1: Performance of `Llama3.1-8B-Base` on MATH after finetuning on increasing proportions of OpenMathInstruct-2.
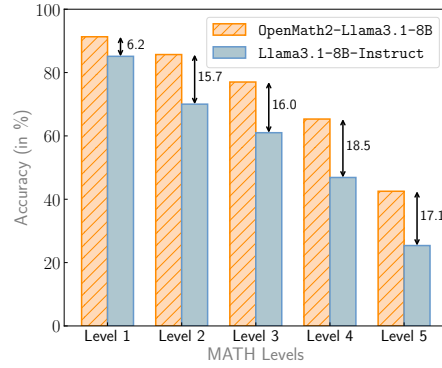

Figure 2: Comparison of `OpenMath2-Llama3.1-8B` and `Llama3.1-8B-Instruct` on accuracy across MATH difficulty levels.

Among open-source alternatives, the recent NuminaMath dataset (Li et al., 2024d) has the largest collection of questions collected from diverse sources. However, its restrictive license—likely due to the use of GPT-4o in data processing and synthesis—limits its broader use. Similarly, other popular math instruction tuning datasets, such as MetaMathQA (Yu et al., 2024) and MathInstruct (Yue et al., 2024a), have also utilized GPT models for data synthesis, which prohibits their usage in non-commercial settings. A notable exception is the OpenMathInstruct-1 (Toshniwal et al., 2024) dataset, one of the biggest open-source math reasoning datasets, where solutions are synthesized using open-weight models. However, OpenMathInstruct-1 has two key limitations. Firstly, its question diversity is limited, since all the questions in the dataset are drawn from the training sets of MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021). Secondly, at the time of its release, there was a sizable gap in the math reasoning capabilities of open and closed-source models. As a result, the dataset underrepresents more challenging problems compared to its GPT-based counterparts (Gou et al., 2024).

The recent emergence of *frontier* open-weight models (Meta-AI, 2024; DeepSeek-AI, 2024) has made it possible to create high-quality, commercially permissible math reasoning datasets. In this paper, we use the recently released Llama3.1 family of models to generate synthetic math instruction tuning (SFT) data, and evaluate the quality of the math reasoning data by finetuning the smaller 8B and 70B base models. To create OpenMathInstruct-2, we conduct careful ablation studies using the MATH dataset to determine design choices that impact the final SFT performance. The highlights of our findings include:

- *Chain-of-Thought (CoT) Solution Format*: Excessive verbosity can be detrimental to the SFT performance. Our proposed CoT format outperforms Llama's CoT format by 3.9% while being 40% shorter in solution length. Using *base model template* (Figure 8 in Appendix) significantly increases the ability of instruct models to follow few-shot examples of our proposed format.

- *Choice of Data Generation Model*: Controlling for the size of the SFT data, the performance on data generated by a strong teacher model surpasses that of data produced by a weaker student model by 7.8%.

- *Robustness of SFT*: With both removing low-quality solutions and introducing them by design, we find SFT performance to be robust to the presence of up-to 20% low-quality data.

- *Impact of Question Diversity*: Controlling for SFT data size, we find that question diversity has a huge positive impact on SFT performance. Increasing the number of unique questions from 1K to 6.5K leads to 10.5% improvement on MATH validation set.

Based on the above findings, we create OpenMathInstruct-2 with data synthesized using `Llama-3.1-405B-Instruct`. To construct this dataset we prompt an LLM to (a) synthesize solutions to the original MATH and GSM8K training set questions and (b) create new question-solution pairs similar to the training set questions. To ensure there is no test set contamination among the synthesized questions, we perform thorough decontamination using the `lm-sys` pipeline (Yang
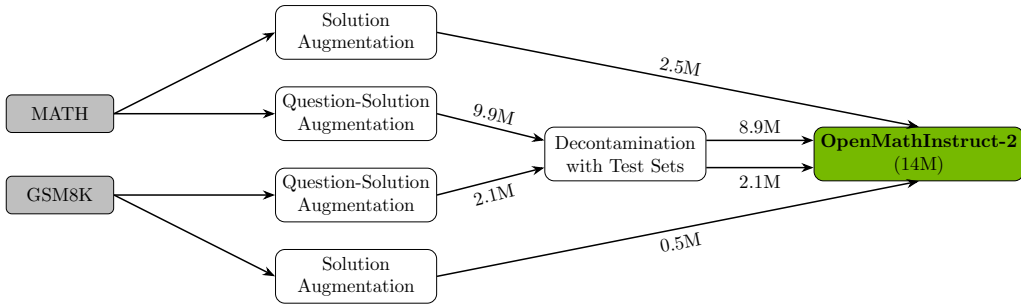
Figure 3: Overview of the data generation pipeline used for OpenMathInstruct-2.

et al., 2023), followed by manual inspection (Section 3.1). Figure 3 provides an overview of the entire dataset construction pipeline. The final dataset consists of 14M question-solution pairs with 600K unique questions, including 592K synthesized questions. Thus, OpenMathInstruct-2 is about 8 times bigger than the previous biggest standalone open-source dataset (Toshniwal et al., 2024).

The high-quality of OpenMathInstruct-2 is illustrated by the strong performance of the finetuned models. The `OpenMath2-Llama3.1-8B model`, which is the `Llama3.1-8B-Base` model finetuned with OpenMathInstruct-2, outperforms `Llama3.1-8B-Instruct` by an absolute 15.9% on MATH with just SFT (see Figure 1 and 2). With a performance of 67.8% on MATH, `OpenMath2-Llama3.1-8B` is one of the strongest sub-10B open-source models.[2] Our best-performing model, `OpenMath2-Llama3.1-70B`, has an accuracy of 71.9% on MATH which outperforms `Llama3.1-70B-Instruct` by 3.9%. To support the open-source efforts, we will release all our fine-tuned models, code and the OpenMathInstruct-2 dataset.

2 DATA: SOLUTION AUGMENTATION

In this section, we focus on the *Solution Augmentation* part of the OpenMathInstruct-2 construction pipeline, shown in Figure 3. We first give a brief overview of how solutions are synthesized for existing questions, and then present ablation studies designed to understand the impact of the different dataset design choices.

2.1 SOLUTION AUGMENTATION PRELIMINARIES

Let $\mathcal{X} = \{(q_i, a_i)\}_{i=1}^N$ represent a typical mathematical reasoning dataset, where q_i and a_i denote the ith question and answer respectively. To synthesize solutions for this dataset, a *teacher* LLM $\mathcal{M}$ is prompted as follows:

$$\mathcal{I}\left(q_1, s_1\right), \ldots, \left(q_K, s_K\right), q'$$

where $\mathcal{I}$ represents the instruction to answer the given math question, $\{q_1, \ldots, q_K\}$ represent K questions representative of the dataset, $\{s_1, \ldots, s_K\}$ represent their respective solutions, and q' represents a question from the training set. Given this prompt, multiple candidate solutions are sampled using $\mathcal{M}$. The high-quality solutions, usually those that lead to the correct answer, along with the prompt question q', are added to the SFT dataset.

2.2 ABLATION STUDIES

In the previous section, we gave an abstract overview of the solution augmentation pipeline. In practice, several design decisions impact the final SFT dataset, such as the solution format of the few-shot examples $\{s_1, \ldots, s_K\}$, the choice of the teacher model $\mathcal{M}$, and the solution filtering mechanism. In this section, we study the impact of these different design choices on the SFT performance to guide the dataset construction.

For these ablation experiments, we use the 1K validation split created from MATH (Hendrycks et al., 2021) training set by Toshniwal et al. (2024). The remaining 6.5K MATH training set problems are

---

[2]We refer to open-weight base models instruction tuned with publicly released data as open-source.

Question

Ben throws four identical darts. Each hits one of four identical dartboards on the wall. After throwing the four darts, he lists the number of darts that hit each board, from greatest to least. How many different lists are possible?

Llama CoT

Step 1: Understand the constraints of the problem
There are four identical darts and four identical dartboards. The task is to list the number of darts that hit each board from greatest to least.

Step 2: Enumerate the possibilities
Given the constraints, we can start by listing the possibilities systematically, starting with the highest number of darts on one board and then distributing the remaining darts across the other boards in decreasing order.

Step 3: List all possible distributions
1. All four darts hit one board: 4, 0, 0, 0
2. Three darts hit one board, and one dart hits another: 3, 1, 0, 0
3. Two darts hit one board, and two darts hit another: 2, 2, 0, 0
4. Two darts hit one board, one dart hits another, and one dart hits yet another: 2, 1, 1, 0
5. One dart hits each board: 1, 1, 1, 1

Step 4: Count the different possibilities
There are 5 different ways to distribute the darts as listed above.

The final answer is: $\boxed{5}$

OpenMath CoT

The number of darts that hit each board must be a non-negative integer, and these numbers must add to 4. Since the dartboards are identical, we can list the numbers from greatest to least.

The possibilities are:

$$4 + 0 + 0 + 0 = 4 + 0 + 0 = 4 + 0 = 4$$
$$3 + 1 + 0 + 0 = 4 + 0 + 0 = 4 + 0 = 4$$
$$2 + 2 + 0 + 0 = 4 + 0 + 0 = 4 + 0 = 4$$
$$2 + 1 + 1 + 0 = 3 + 1 + 0 = 4 + 0 = 4$$
$$1 + 1 + 1 + 1 = 2 + 1 + 1 = 3 + 1 = 4$$
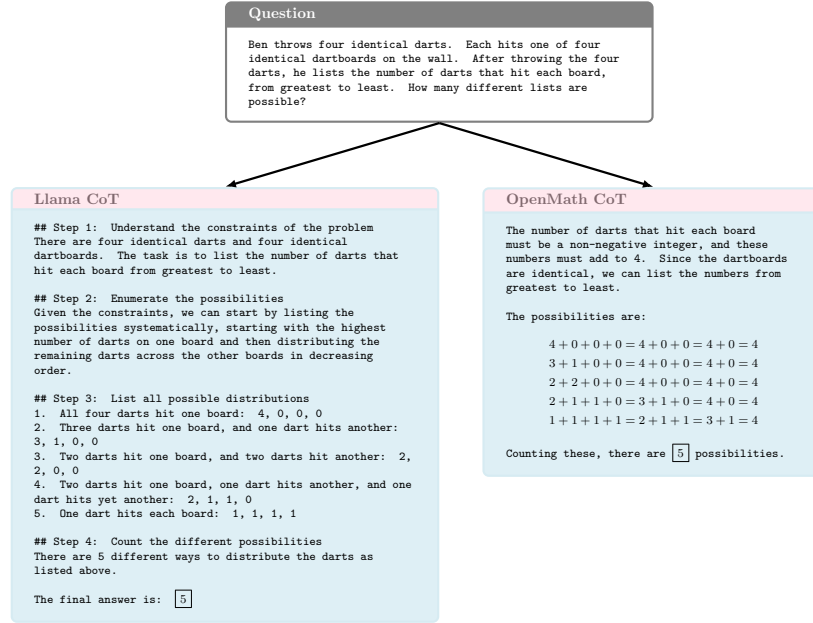
Counting these, there are $\boxed{5}$ possibilities.

Figure 4: A sample solution in the Llama CoT format vs. the OpenMath CoT format.

used to create the SFT dataset. The solutions are generated using nucleus sampling (Holtzman et al., 2020) with a temperature of 1.0 and top-p of 0.95. The `Llama3.1-8B-Base` model is used as the *student* model in all the ablation experiments. For SFT, the model is trained for 4 epochs, with a batch size of 256, using the AdamW optimizer (Loshchilov and Hutter, 2019) with a constant learning rate of 5e-6 and a weight decay of 1e-2. To account for the variance in performance across runs, we report the performance averaged across 4 runs.

Data Downsampling For efficiency or experiment design reasons, we sometimes need to downsize an SFT dataset to a specific size or to match another SFT dataset in ablation experiments. We introduce the concept of *coverage* and the two downsampling operations used in the paper.

Coverage of a SFT dataset $\mathcal{D} = \{(q_i, s_i)\}_{i=1}^{T}$ synthesized using dataset $\mathcal{X} = \{(q_i, a_i)\}_{i=1}^{N}$ is the fraction of questions in $\mathcal{X}$ with at least one solution in $\mathcal{D}$:

$$\text{Coverage}(\mathcal{D}, \mathcal{X}) = \frac{|\{q : (q, s) \in \mathcal{D}\}|}{|\{q : (q, a) \in \mathcal{X}\}|}$$

Fair Downsampling is a question-dependent downsampling method introduced by Toshniwal et al. (2024). Due to the varying difficulty of questions, the representation of "easier" ones can often dominate an SFT dataset, as generating high-quality solutions for them is "easier". The goal of *fair* downsampling is to sample question-solution pairs from the original SFT dataset in a way that ensures all questions are as equally represented in the downsampled dataset as possible.

Matching Coverage: The different design choices explored in the ablation studies result in SFT datasets of varying sizes. However, to compare the quality of the datasets, we want to control for the dataset size. To this end, we introduce the *Matching Coverage* operation, where SFT datasets are matched at the level of questions. Put simply, after matching coverage, the number of unique questions as well as the number of solutions for each individual question in two dataset is the same.

Formally, suppose we're given two SFT datasets $\mathcal{D}_1$ and $\mathcal{D}_2$. Let $Q(\mathcal{D}_1)$ represent the set of unique questions in $\mathcal{D}_1$:

$$Q(\mathcal{D}_1) = \{q \mid (q, s_1) \in \mathcal{D}_1\}$$

The set of common questions in $\mathcal{D}_1$ and $\mathcal{D}_2$ is given by:

$$Q_{\text{match}} = Q(\mathcal{D}_1) \cap Q(\mathcal{D}_2)$$

4

Table 1: Comparison of Llama and OpenMath CoT formats on MATH validation accuracy and average solution length measured in number of tokens.

	MATH Validation Accuracy	Mean Solution Length
Llama CoT	40.6 ± 0.6	331.3
OpenMath CoT	44.5 ± 0.8	237.0

Let $N(\mathcal{D}, q)$ represent the number of solutions of question q in dataset $\mathcal{D}$. In the matching coverage version of the datasets:

$$N_{\text{match}}(q) = \min(N(\mathcal{D}_1, q), N(\mathcal{D}_2, q))$$

for each question $q \in Q_{\text{match}}$, $N_{\text{match}}(q)$ solutions are sampled from the respective datasets.

This covers the two downsampling methods used in this paper: *Fair Downsampling* and *Matching Coverage*. Next, we will describe the ablation experiments.

2.2.1 SOLUTION FORMAT

Finetuning with synthetic chain-of-thought (CoT) solutions (Nye et al., 2021; Wei et al., 2022; Sprague et al., 2024) has been the key to strong performances of small models on math reasoning tasks (Yu et al., 2024; Toshniwal et al., 2024; Meta-AI, 2024). We find the Llama's CoT format to be quite verbose,[3] and propose an alternate CoT format, *OpenMath CoT*, which is detailed as well but less verbose. Figure 4 shows a sample solution in the two CoT formats.

To compare the two CoT formats, we generate SFT data using the `Llama3.1-405B-Instruct` model. For generating solutions in the Llama CoT format we simply use the zero-shot prompt setup as the model was trained on those kinds of solutions. However, even when prompting the model with few-shot OpenMath CoT solutions, a substantial number of generations – 57% in our experiment – still follow the Llama CoT format. This tendency of *aligned* models reverting to their trained behavior when encountering inputs seen during training has also been observed in prior work (Min et al., 2022). We find an interesting workaround to this issue by dropping the special tokens used by Llama-Instruct models. Prompting the model with the "base" template leads to a dramatic increase in adherence to the OpenMath CoT format and reduces the Llama CoT format generations to only 0.1%. See Appendix A.1 for the prompt and more details.

With 64 solutions sampled per question, the zero-shot setup results in about 30% more solutions than the few-shot prompt setup (350K vs 268K). To control for the confounding factor of SFT data size, we perform the Matching Coverage operation over the two datasets which reduces the final SFT dataset to 260K question-solution pairs. Table 1 shows that the OpenMath CoT format is 40% less verbose than the Llama CoT format and also results in a better SFT performance. All experiments presented henceforth use the OpenMath CoT format.

2.2.2 CHOICE OF TEACHER MODEL

Prior work has shown that with repeated sampling, even weak models can match or outperform much stronger/bigger models (Li et al., 2024b; Brown et al., 2024). In fact, for a fixed compute budget, a weaker model can be a better choice for a teacher model (Bansal et al., 2024). But data synthesis is a one-time expense and a small portion of the overall compute budget of training LLMs (Villalobos and Atkinson, 2023). We instead ask the following question: *Can a student model learn better from its own generated solutions vs solutions generated by a strong teacher model when matching the SFT data coverage?*

In this ablation, we compare `Llama3.1-8B-Base` and `Llama3.1-405B-Instruct` as teacher models. We sample solutions using the two models and perform the Matching Coverage operation to match the final SFT datasets precisely. The SFT results presented in Table 2 show that even when controlling for the SFT data size, `Llama3.1-405B-Instruct` is a far superior data generation model.

---

[3]`https://huggingface.co/datasets/meta-llama/Meta-Llama-3.1-8B-Instruct-evals/viewer/Meta-Llama-3.1-8B-Instruct-evals__math__details`

Table 2: `Llama3.1-8B-Base` vs. `Llama3.1-405B-Instruct` as data generation models.

	MATH Validation Accuracy	Mean Solution Length
Llama3.1-8B-Base	30.1 ± 0.6	205.7
Llama3.1-405B-Instruct	37.9 ± 0.6	180.2

Our preliminary analysis suggests that the reason is weaker models generate more *noisy solutions* that use incorrect reasoning yet end up with the right answer and, ultimately, part of the SFT dataset (Appendix B). We leave a more detailed analysis regarding this for future work. Next, we investigate the impact of these *noisy solutions* among solutions generated by `Llama3.1-405B-Instruct`.

2.2.3 IMPACT OF LOW-QUALITY SOLUTIONS

Table 3: SFT performance on the MATH validation set with various filtering strategies to remove solutions with incorrect reasoning.

Filtering Strategy	Data Size	MATH Validation Accuracy
Unfiltered	128K	43.6 ± 1.7
LLM-as-a-Judge: Prompt 1	113K	43.6 ± 0.1
LLM-as-a-Judge: Prompt 2	116K	43.0 ± 0.8
Nemotron-4-340B-Reward: Helpfulness ≥ 3	118K	43.8 ± 0.4
Nemotron-4-340B-Reward: Correctness ≥ 3	120K	43.1 ± 0.4

Data quality plays an important role in the accuracy of LLMs (Jain et al., 2024). We explore the impact of data quality on the final SFT performance in our setup. First, we employ automated LLM-based methods to filter out solutions that, despite reaching the correct answer, use incorrect reasoning. Second, we investigate the effects of intentionally incorporating incorrect solutions into the SFT dataset.

Removing Low-Quality Solutions. Synthetic solutions produced in our pipeline may include examples where the intermediate steps are incorrect, yet still lead to the right final answer. For simplicity, we refer to these instances as "low-quality" data. In this section, we will discuss how we identify and remove low-quality data, followed by an investigation into its impact on the SFT performance.

We employ two methods to identify low-quality data: LLM-as-a-Judge and reward model. In the LLM-as-a-Judge approach, we design two prompts for the `Llama3.1-405B-Instruct` to determine whether the generated solutions contain incorrect intermediate steps, providing a binary outcome (see Appendix D.3 for the prompts). For the reward model labeling method, we use Nemotron-4-340B-Reward (Wang et al., 2024) to evaluate the quality of the generated solutions based on factors like helpfulness (the overall usefulness of the response to the prompt) and correctness (the inclusion of all relevant facts without errors). Helpfulness and correctness are rated on a scale from 0 to 4, where a higher score indicates better data quality. For the reward model filtering, we used a threshold of 3 based on small-scale tuning experiments.

To determine the impact of filtering low-quality data on the SFT performance, we use a 128K-sized fair downsampled SFT dataset. We call this *Unfiltered* data and use a model trained on it as a baseline.

Table 3 presents the statistics of data remaining with different filtering approaches, and the corresponding SFT performance. The proportion of data filtered by the different methods ranges from 6% to 12%, a non-negligible fraction of the overall data. [4] Yet none of the filtering strategies give any meaningful gain over the baseline *Unfiltered* model. This means that either SFT is robust to the presence of up to 10% of low-quality solutions or our filtering is not accurate enough. We investigate this question next.

---

[4] Our manual analysis of 20 examples identified by the two approaches suggests that approximately 60% of the solutions are indeed incorrect.

(a) Adding wrong-answer solutions.
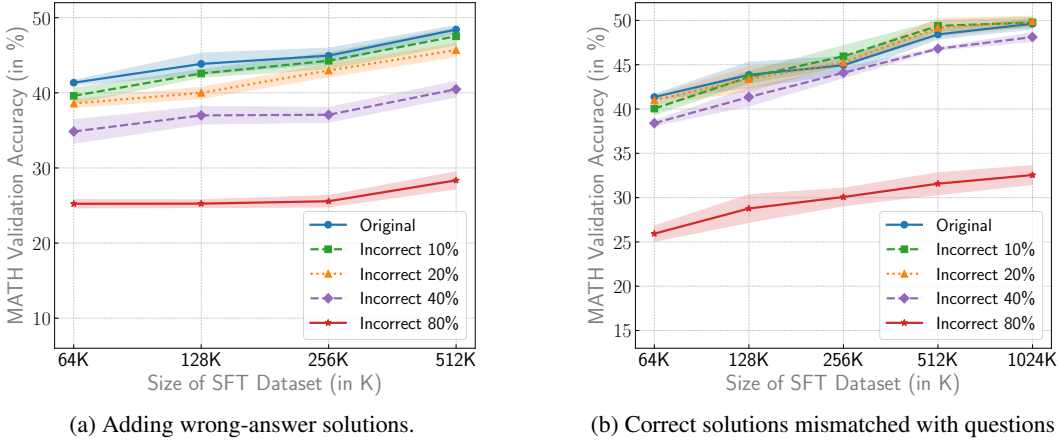
(b) Correct solutions mismatched with questions

Figure 5: Impact of low-quality solutions on the SFT performance.

Adding Low-Quality Solutions. In the previous section, we see that filtering low-quality solutions generated by a strong model such as `Llama3.1-405B-Instruct` leads to almost the same or worse SFT performance in comparison to no filtering. While our manual analysis suggests that most of the filtered out solutions were indeed using incorrect reasoning, the automatic filtering approaches are far from perfect and it's hard to gauge the impact of filtering out correct solutions which have been classified as incorrect.

To remove the effect of potentially inaccurate filtering, we can instead study the impact of explicitly adding low-quality/incorrect solutions on the SFT performance. We consider two strategies of adding "bad" solutions:

1. **Wrong-answer Solutions**: By incorporating solutions generated by the teacher LLM, which were excluded during the creation of the SFT dataset due to not arriving at the ground truth answer.

2. **Incorrect Pairing**: By shuffling some of the question-solution pairs in the SFT dataset, such that the correct solutions are paired with unrelated questions.

For both these strategies, we experiment with varying the proportion of such incorrect solutions from {10%, 20%, 40%, 80%}. We also vary the SFT data size from {64K, 128K, 256K, 512K, 1024K} to study the impact on SFT performance at different data scales[5].

Figure 5 presents the impact of incorrect solutions on the SFT performance at varying data sizes. From both the plots we see that the model performance suffers little to no performance degradation with as much as 20% incorrect solutions at data scales $\geq$ 256K. Among the two strategies, we see that the model is especially robust to "Incorrect Pairing" with strong performance even with 40% incorrect solutions.

Based on these results we conclude that models are indeed robust to the presence of up-to 20% of low-quality solutions during SFT and extensive data filtering at this stage has limited gains.

2.2.4 IMPACT OF QUESTION DIVERSITY

To investigate the impact of question diversity on SFT performance, we construct finetuning datasets with 256K question-solution pairs with the number of unique questions varying from {1K, 2K, 4K, 6.5K}. Figure 6 shows a clear trend that the SFT performance improves with an increase in the number of unique questions, with a drop of more than 10 points when the number of unique questions is limited to 1K. This result highlights the potential of generating new questions, and we describe the Question-Solution Augmentation pipeline next.

---

[5]For the "Wrong-answer Solutions" setting, we were not able to run the experiments for 1024K data size because the `Llama3.1-405B-Instruct` model makes few mistakes on the MATH training set.
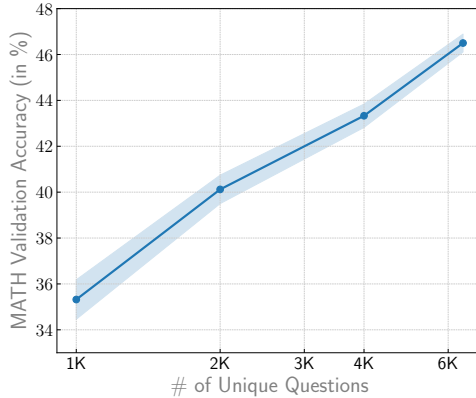
Figure 6: Impact of question diversity on MATH validation accuracy.

3 DATA: QUESTION-SOLUTION AUGMENTATION

In this section, we describe the Question-Solution Augmentation component of the OpenMathInstruct-2 construction pipeline, illustrated in Figure 3. This process consists of two stages: (i) question augmentation, and (ii) solution augmentation.

For question augmentation, we utilize the training splits of MATH and GSM8K as seed datasets to generate new questions. We use simple few-shot prompting showing 5 examples of original questions and the new questions written by us that are similar in some aspect. We do not add explicit instructions to increase difficulty or add new conditions, instead relying on the inherent variance of the nucleus sampling that we use to generate new problems. After filtering out syntactically ill-formed questions, we check the generated questions for potential contamination with test sets of evaluation benchmarks, described in detail in the next section. To generate solutions for the new synthesized questions, we use the solution augmentation pipeline from Section 2.1, generating 32 solutions for each question with a temperature of 0.7. Since the newly synthesized questions don't have ground-truth answers to filter solutions, we instead use majority voting among the 32 generations as a proxy for the ground-truth answer. For more details on question-solution augmentation, see Appendix C.

3.1 LLM DECONTAMINATION

It has been noted that many widely used benchmarks and datasets suffer from data contamination, where information from the test set unintentionally leaks into the training data (Yang et al., 2023). This can result in an overly optimistic assessment of the model's performance. The most commonly used methods, such as n-gram overlap and embedding similarity search, are susceptible to simple variations in test data (e.g., paraphrasing, translation), allowing rephrased samples to bypass these basic detection techniques easily.

We adopt the approach suggested by Yang et al. (2023) to remove all potential paraphrases of evaluation benchmark questions from the synthesized questions. In our setup, we use the test sets of four evaluation benchmarks, namely GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), AMC 2023 (AMC 2023, 2023), and AIME 2024 (AIME 2024, 2024).

The LLM-based decontamination process consists of two main steps. First, for each synthesized question, use embedding similarity search to identify the top-k most similar test examples from all benchmark datasets. Second, create question pairs by matching the synthesized question with each of these top-k test examples. An advanced LLM then evaluates whether any of these pairs are paraphrases via zero-shot prompting. To mitigate any positional bias, we generate two pairs for each match: one in which the synthesized question appears first and another in which the test set question is presented first. If any of the $2k$ pair is determined to be a paraphrase, the synthesized question is removed.

We use a popular *Sentence Transformer* model for embedding,[6] and `Llama3.1-405B-Instruct` for paraphrase detection (details on the prompt are provided in Appendix D.4). In our experiment,

---

[6]https://huggingface.co/sentence-transformers/multi-qa-MiniLM-L6-cos-v1

Table 4: Comparison of our *OpenMath2-Llama* models with other open-weight and open-source models without tool usage. Open-weight base models finetuned with publicly released data are considered as open-source for the purposes of this table.

Category	Params	Model	GSM8K	MATH	AMC 2023	AIME 2024	Omni-MATH[7]
Open Weight	< 10B	Qwen2.5-Math-7B-Instruct (Yang et al., 2024)	95.2	83.6	25/40	5/30	32.3
		Mathstral-7B (Mistral AI, 2024)	77.1	56.6	-	-	-
		Llama3.1-8B-Instruct (Meta-AI, 2024)	84.2	51.8	9/40	2/30	12.7
Open Source		NuminaMath-7B-CoT (Li et al., 2024d)	75.4	55.2	11/40	0/30	-
		OpenMath2-Llama3.1-8B (ours)	91.7	67.8	16/40	3/30	22.0
		+ maj@256	94.1	76.1	23/40	3/30	24.6
Open Weight	10 to 100B	DS-Coder-V2-Lite-Instruct (DeepSeek-AI et al., 2024)	86.4	61.8	-	0/30	19.7
		Qwen2.5-Math-72B-Instruct (Yang et al., 2024)	95.9	85.0	28/40	9/30	36.3
		Llama3.1-70B-Instruct (Meta-AI, 2024)	95.8	67.9	19/40	6/30	19.0
Open Source		NuminaMath-72B-CoT (Li et al., 2024d)	91.4	68.0	21/40	1/30	28.4
		OpenMath2-Llama3.1-70B (ours)	94.9	71.9	20/40	4/30	23.1
		+ maj@256	96.0	79.6	24/40	6/30	27.6

we use $k = 5$, which results in 10 LLM inference calls for each generated question. To emphasize the importance of using an LLM in the decontamination pipeline, we provide multiple examples of questions flagged as contaminated that cannot be found via n-gram matching (see Table 10 in the Appendix). Overall, our decontamination pipeline removes about 50K questions out of the 569K new questions synthesized (569K ⟶ 519K).

4 RESULTS

Training Details. All the models are trained with a batch size of 512, using the AdamW optimizer (Loshchilov and Hutter, 2019) with a constant learning rate of 2e-5 and a weight decay of 1e-2. For the 8B model, we train the model on 1M, 2M, and 5M fair downsampled versions of OpenMathInstruct-2 to understand the impact of the data scaling. Due to computational constraints, we train the 70B model only on the 5M subset with a learning rate of 1e-5. The models are trained for 2 epochs, and we save 6 equally spaced checkpoints during the training runs, which are averaged to create the final model (See Appendix A.4 for performance gains with checkpoint averaging).

Evaluation Details. We evaluate our models on a set of common benchmarks that consists of GSM8K (1.3K examples), MATH (5K examples), AMC 2023 (40 examples), AIME 2024 (30 examples), and Omni-MATH (4.4K examples) (Omni-Math, 2024). These datasets cover a broad spectrum of difficulty levels, ranging from grade school mathematics to advanced competition problems. Unless noted otherwise, all fine-tuned models are assessed in a zero-shot setting with both greedy decoding and majority voting out of 256 sampled solutions with temperature of 0.7 (Wang et al., 2022).

We use GPT-4o (OpenAI, 2023) as a judge to compare the ground truth answers with those predicted by our models (the detailed prompt is provided in Appendix D.5).

Impact of Data Scaling. Figure 1 plots the performance on the MATH test set with the increase in SFT data size. With even the 1M fair-downsampled version of OpenMathInstruct-2, the final model easily outperforms `Llama3.1-8B-Instruct` and `NuminaMath-7B-CoT`. We observe a consistent gain with an increase in data size, and even at 14M dataset size, we see no signs of saturation in performance gains.

Final Results. Table 4 presents the results for top-performing, open-weight and open-source models (without tool use). The `OpenMath2-Llama3.1-8B` model, which is finetuned on the full OpenMathInstruct-2 dataset, outperforms or matches `Llama3.1-8B-Instruct` on all the math reasoning benchmarks. Among the open-source models, we outperform the recently released `NuminaMath-7B-CoT` on all benchmarks as well. Finally, among all the presented models, the `OpenMath2-Llama3.1-8B` is second only to the `Qwen2.5-Math-7B-Instruct`, which has

---

[7]Omni-MATH dataset was released after we finished training our models, so we didn't use it during decontamination. After checking for contamination, we found that about 1.4% of the test set questions are part of our training data.

been trained on more than a trillion synthetically generated math reasoning tokens, and starts with a base model, `Qwen2.5-Math`, which is about 35% better than `Llama3.1-8B-Base`. [8]

The `OpenMath2-Llama3.1-70B` is our strongest performing model which is the `Llama3.1-70B-Base` model finetuned on the 5M fair downsampled subset of OpenMathInstruct-2. While our 8B model demonstrates strong accuracy gains compared to other LLMs of similar size, the 70B model only shows improvements on a subset of benchmarks. We hypothesize that our data blend or solution format might be more suited for weaker models, since we made all of the design decisions based on the 8B model accuracy on validation subsets.

5 RELATED WORK

In recent years, significant progress has been made in developing datasets to enhance mathematical reasoning abilities of LLMs. NuminaMath (Li et al., 2024d) contains a collection of 860K pairs of competition-level math problems and solutions, annotated with chain-of-thought traces (Wang et al., 2023). Skywork-MathQA (Zeng et al., 2024) collects 2.5M question-solution pairs, incorporating three different augmentation techniques and a diverse seed problem set. MuggleMath (Li et al., 2024c) is created by complicating and diversifying queries, as well as sampling multiple reasoning paths from existing datasets. MetaMathQA (Yu et al., 2024) introduced a dataset with 395K entries created by bootstrapping questions from MATH and GSM8K, employing techniques such as semantic rephrasing, self-verification, and backward reasoning. MAmmoTH2 (Yue et al., 2024b) introduced a paradigm for efficiently extracting 10 million naturally occurring instruction data points from pre-training web corpora, enhancing LLM reasoning and improving benchmark performance without the need for in-domain training. Li et al. (2024a) expanded the MATH dataset to 480K and the GSM8K dataset to 960K by generating both questions and CoT-based solutions, resulting in significant accuracy improvements for fine-tuned models.

Tool-integrated methods for math problem-solving have also become prevalent. Chen et al. (2023) pioneered the Program of Thoughts (PoT) approach, combining text and programming language statements to arrive at solutions. Building on similar concepts, other datasets have been developed. For instance, OpenMathInstruct-1 (Toshniwal et al., 2024) introduced a math instruction tuning dataset of 1.8 million examples, synthesizing code-interpreter solutions for GSM8K and MATH benchmarks. InfinityMATH (Zhang et al., 2024a) developed a scalable instruction tuning dataset for programmatic mathematical reasoning, consisting of 100K data points.

Similar to prior work, we also leverage CoT-based solutions and question augmentation to construct a novel dataset. Yet our approach distinguishes itself in several important ways: (a) we leverage open-weight models instead of proprietary closed-source LLMs allowing us to release the dataset under a permissive license; (b) we offer novel insights into the impact of low-quality data, and the design of solution format; (c) we ensure our results are accurate by performing a comprehensive decontamination process using an LLM-based pipeline that can detect rephrased variations of test set questions.

6 CONCLUSION

Recent advances in LLM mathematical reasoning have mostly been *closed-source* since instruction tuning data is often not shared or has restrictive license. In this paper we contribute towards *open-source* progress by sharing the OpenMathInstruct-2 dataset and all the code necessary to reproduce our work. Besides releasing high-performing models and data, we also conduct detailed ablations that advance our understanding of how to best construct such datasets. In summary, we show that: a) Not all CoT formats are equally effective, and longer solutions are not necessarily better; b) Performance on data generated by a strong teacher model surpasses that of equally-sized data produced by a weaker student model; c) Data filtering has limited utility for math reasoning datasets as models are quite robust to the presence of incorrect solutions during SFT; and d) Training on a diverse set of questions is crucial, but proper decontamination has to be performed to ensure the benchmark evaluations accurately represent model strengths.

---

[8] We are unsure of the n-gram based data contamination protocol followed by `Qwen2.5-Math` given its obvious weakness in detecting paraphrases.

REFERENCES

AIME 2024. `https://artofproblemsolving.com/wiki/index.php/2024_AIME_I`, 2024.

Rachith Aiyappa, Jisun An, Haewoon Kwak, and Yong-yeol Ahn. Can we trust the evaluation on ChatGPT? In *3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, 2023.

AMC 2023. `https://github.com/QwenLM/Qwen2.5-Math/blob/main/evaluation/data/amc23/test.jsonl`, 2023.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. Llemma: An Open Language Model for Mathematics. In *ICLR*, 2024.

Hritik Bansal, Arian Hosseini, Rishabh Agarwal, Vinh Q. Tran, and Mehran Kazemi. Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling, 2024.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large Language Monkeys: Scaling Inference Compute with Repeated Sampling, 2024.

Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling Synthetic Data Creation with 1,000,000,000 Personas, 2024.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. *TMLR*, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, 2021.

DeepSeek-AI. DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model, 2024.

DeepSeek-AI, Qihao Zhu, Daya Guo, Zhihong Shao, Dejian Yang, Peiyi Wang, Runxin Xu, Y. Wu, Yukun Li, Huazuo Gao, Shirong Ma, Wangding Zeng, Xiao Bi, Zihui Gu, Hanwei Xu, Damai Dai, Kai Dong, Liyue Zhang, Yishi Piao, Zhibin Gou, Zhenda Xie, Zhewen Hao, Bingxuan Wang, Junxiao Song, Deli Chen, Xin Xie, Kang Guan, Yuxiang You, Aixin Liu, Qiushi Du, Wenjun Gao, Xuan Lu, Qinyu Chen, Yaohui Wang, Chengqi Deng, Jiashi Li, Chenggang Zhao, Chong Ruan, Fuli Luo, and Wenfeng Liang. DeepSeek-Coder-V2: Breaking the Barrier of Closed-Source Models in Code Intelligence, 2024.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. ToRA: A Tool-Integrated Reasoning Agent for Mathematical Problem Solving. In *ICLR*, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset. In *NeurIPS Datasets and Benchmarks*, 2021.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration. In *ICLR*, 2020.

Naman Jain, Tianjun Zhang, Wei-Lin Chiang, Joseph E. Gonzalez, Koushik Sen, and Ion Stoica. LLM-Assisted Code Cleaning For Training Accurate Code Generators. In *ICLR*, 2024.

Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. Common 7B Language Models Already Possess Strong Math Capabilities. *arXiv preprint arXiv:2403.04706*, 2024a.

Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. Common 7B Language Models Already Possess Strong Math Capabilities, 2024b.

Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. MuggleMath: Assessing the Impact of Query and Response Augmentation on Math Reasoning. In *ACL*, 2024c.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. NuminaMath: The largest public dataset in AI4Maths with 860k pairs of competition math problems and solutions, 2024d.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2019.

Meta-AI. The Llama 3 Herd of Models, 2024.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? In *EMNLP*, 2022.

Mistral AI. `https://mistral.ai/news/mathstral/`, 2024.

NVIDIA. Nemotron-4 340B Technical Report, 2024.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show Your Work: Scratchpads for Intermediate Computation with Language Models, 2021.

Omni-Math. `https://omni-math.github.io/`, 2024.

OpenAI. Gpt-4 technical report, 2023. URL `https://openai.com/research/gpt-4`.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. In *ACL*, 2024.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. To CoT or not to CoT? Chain-of-thought helps mainly on math and symbolic reasoning, 2024.

Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. OpenMathInstruct-1: A 1.8 Million Math Instruction Tuning Dataset. In *NeurIPS Datasets and Benchmarks*, 2024.

Pablo Villalobos and David Atkinson. Trading Off Compute in Training and Inference, 2023. URL `https://epochai.org/blog/trading-off-compute-in-training-and-inference`.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*, 2023.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. HelpSteer2: Open-source dataset for training top-performing reward models, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *NeurIPS*, 2022.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement, 2024.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. Rethinking Benchmark and Contamination for Language Models with Rephrased Samples, 2023.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. In *ICLR*, 2024.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MAmmoTH: Building math generalist models through hybrid instruction tuning. In *ICLR*, 2024a.

Xiang Yue, Tuney Zheng, Ge Zhang, and Wenhu Chen. MAmmoTH2: Scaling Instructions from the Web. *arXiv preprint arXiv:2405.03548*, 2024b.

Liang Zeng, Liangjun Zhong, Liang Zhao, Tianwen Wei, Liu Yang, Jujie He, Cheng Cheng, Rui Hu, Yang Liu, Shuicheng Yan, Han Fang, and Yahui Zhou. Skywork-Math: Data Scaling Laws for Mathematical Reasoning in Large Language Models – The Story Goes On, 2024.

Bo-Wen Zhang, Yan Yan, Lin Li, and Guang Liu. InfinityMATH: A Scalable Instruction Tuning Dataset in Programmatic Mathematical Reasoning. *arXiv preprint arXiv:2408.07089*, 2024a.

Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Dylan Slack, Qin Lyu, Sean Hendryx, Russell Kaplan, Michele Lunati, and Summer Yue. A Careful Examination of Large Language Model Performance on Grade School Arithmetic, 2024b.

A MISCELLANEOUS

A.1 GENERATING SOLUTION IN OPENMATH COT FORMAT

Instruct Prompt Template

```
<|begin_of_text|><|start_header_id|>user<|end_header_id|>

FEW-SHOT PROMPTS

Question:
{question}<|eot_id|><|start_header_id|>assistant<|end_header_id|>

{generation}
```

Figure 7: Typical *instruct* prompt template used with `Llama-Instruct` models.

Base Prompt Template

```
<|begin_of_text|>FEW-SHOT PROMPTS

Question:
{question}

My solution:
{generation}
```

Figure 8: *Base* prompt template where we drop the special tokens for marking roles when using the `Llama-Instruct` models.

When we prompt the `Llama3.1-405B-Instruct` model with few-shot examples in OpenMath CoT format from Appendix D.1 in tandem with the *instruct prompt*, shown in Figure 7, almost 57% of the generated solutions are in the Llama CoT format on which the model is most likely trained on.[9] We find that dropping the Llama special tokens for marking roles in the prompt, as shown in Figure 8, results in much better adherence to our proposed few-shot prompt with only 0.1% generations in the Llama CoT format.

A.2 POST-PROCESSING

We remove or modify solutions based on the following criteria:

- Remove solutions with multiple `\boxed` entries.
- Remove prefix `My Solution:` from solutions.
- Truncate the solution till the first sentence with `\boxed`.
- Remove incorrect arithmetic calculations.
- Split complex arithmetic calculations to step-by-step calculations to make it easier for the model to generate.
- Remove solutions longer than 1024 `Llama3.1` tokens.
- Remove solutions with less than 200 characters.

Table 5: Composition of OpenMathInstruct-2

Dataset	Approach	# of Unique Ques.	# of Unique Ques.-Sol. Pairs
GSM8K	Solution Augmentation	7.4K	0.46M
GSM8K	Question-Solution Augmentation	73.6K	2.11M
MATH	Solution Augmentation	7.4K	2.46M
MATH	Question-Solution Augmentation	519.1K	8.94M
Total	-	607.3K	13.97M

A.3 COMPOSITION OF OPENMATHINSTRUCT-2

Table 5 represents the composition of OpenMathInstruct-2. The dataset consists of about 592K new synthetically-generated questions which contribute about 11M new question-solution pairs.

A.4 CHECKPOINT AVERAGING
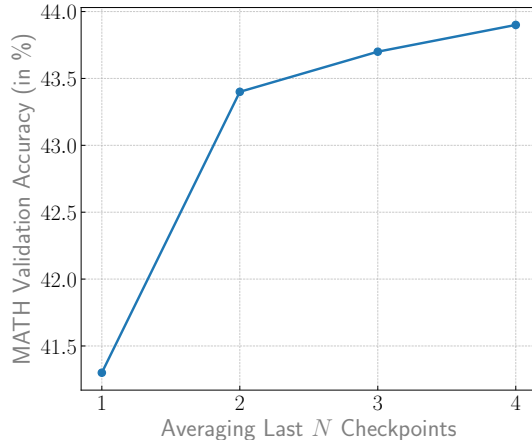
Figure 9: MATH Validation accuracy as a function of the final checkpoint being an average of the last N checkpoints.

We have found consistent gains in our setup with checkpoint averaging. Figure 9 shows a gain of more than 2% for one of our ablation runs when the final checkpoint is created using the average of the last 4 checkpoints in comparison to using only the last checkpoint.

B PERFORMANCE COMPARISON BETWEEN DIFFERENT TEACHER MODELS

In this section, we explore the impact of low-quality data produced by two distinct teacher models: `Llama3.1-8B-Base` and `Llama3.1-405B-Instruct`. To identify low-quality data, we employ the same methods outlined in Section 2.2.3, specifically, LLM-as-a-judge and reward model labeling.

For the teacher model `Llama3.1-8B-Base`, we generated 128K data samples using the same configuration as `Llama3.1-405B-Instruct`, with the MATH dataset serving as the seed. We ensured that all solutions produced led to the correct final answer, and restricted the maximum token length of generated solutions to 1024. Data statistics and SFT performance are summarized in Table 6.

---

[9]`https://huggingface.co/datasets/meta-llama/Meta-Llama-3.`
`1-8B-Instruct-evals/viewer/Meta-Llama-3.1-8B-Instruct-evals__math_`
`_details`

Table 6: Performance of the SFT Llama3.1-8B-Base model on the MATH validation set after applying different filtering strategies to remove poor-quality data from two-choice teacher models: 8B-Base and 405B-Instruct. Results for the 405B-Instruct model are averaged over 4 runs, while the 8B-Base results are based on a single run.

Teacher model	Filtering Strategy	Data Size	MATH Validation Accuracy
405B-Inst	Unfiltered	128K	43.6 ± 1.7
	LLM-as-a-Judge: Prompt 1	113K	43.4 ± 0.1
	LLM-as-a-Judge: Prompt 2	116K	43.0 ± 0.8
	Nemotron-4-340B-Reward: Helpfulness ≥ 3	118K	43.7 ± 0.4
	Nemotron-4-340B-Reward: Correctness ≥ 3	120K	43.1 ± 0.4
8B-Base	Unfiltered	128K	29.8
	LLM-as-a-Judge: Prompt 1	70K	30.3
	LLM-as-a-Judge: Prompt 2	72K	29.3
	Nemotron-4-340B-Reward: Helpfulness ≥ 3	42K	28.1
	Nemotron-4-340B-Reward: Correctness ≥ 3	49K	30.5

The percentage of low-quality data generated by the `Llama3.1-8B-Base` teacher model, when applying different filtering strategies, ranged from 45% to 67%. This is notably higher than the percentage observed with the `Llama3.1-405B-Instruct` model, as expected. More advanced teacher models, like `Llama3.1-405B-Instruct`, generally produce higher-quality data.

The SFT performance of the student model `Llama3.1-8B-Base` remained relatively stable across the various filtering strategies, regardless of whether the teacher was `Llama3.1-8B-Base` or `Llama3.1-405B-Instruct`. However, the overall performance was consistently higher when `Llama3.1-405B-Instruct` was used as the teacher. This observation aligns with the findings discussed in Section 2.2.3, which highlight that SFT performance experiences minimal to no degradation, even when a significant portion of the training data is noisy.

Finally, Table 7 and Table 8 present low-quality solutions identified by the two methods for `Llama3.1-8B-Base` and `Llama3.1-405B-Instruct` respectively.

Table 7: Sample solutions generated by `Llama3.1-8B-Base` identified as having **incorrect** reasoning steps. We highlight in red incorrect judgements made by the models.

Strategy	Judgement correctness	Question	Solution
LLM-as-a-Judge	**Model correctly identifies incorrect reasoning**	The faces of an octahedral die are labeled with digits 1 through 8. What is the probability, expressed as a common fraction, of rolling a sum of 15 with a pair of such octahedral dice?	We need to find the probability of summing 15 with a pair of dice rolls. For each die roll, there are 8 possible outcomes, which are $\{1,2,3,4,5,6,7,8\}$. To calculate the total number of possible outcomes, we find $8 \times 8 = 64$. (Correct reasoning) Now we need to find how many rolls lead to a sum of 15. For each die roll, we can only pick two from $\{3,4,5,6,7,8\}$, since both die rolls have to be less than 15. (Incorrect reasoning) We can pick any combination of two from the set of six, which is $\binom{6}{2} = 15$. There are two ways to roll 15: $(3, 12)$ and $(4, 11)$. (Incorrect reasoning) Now find the probability (Correct final answer): $$\text{probability} = \frac{\text{number of successful rolls}}{\text{total number of rolls}} = \frac{2}{64} = \boxed{\frac{1}{32}}$$
LLM-as-a-Judge	**Model labels as incorrect, but the solution is correct**	In a physical education class, students line up in four rows to do exercises. One particular class has over 30 students, with three rows having the same number of students and one row having one more student than the other three rows. What is the smallest possible class size for this physical education class?	Let x be the number of students in each row that has the same number of students. Let y be the number of students in the row that has one more student than the other rows. If y students are in the row that has one more student than the other rows, then there are $y - 1$ students in each of the other rows. So the total number of students is (Correct reasoning) $$(y-1)+(y-1)+(y-1)+y = 3y-3+y = 4y-3$$ We need the total number of students to be greater than 30. Thus, we have (Correct reasoning) $$4y-3 > 30 \Rightarrow 4y > 33 \Rightarrow y > 33:4 \Rightarrow y > 8$$ With the constraints $y \geq 9$ and y is an integer, the smallest possible value of y is $y = 9$. (Correct reasoning) Thus the smallest possible class size is $4y - 3 = 4(9) - 3 = \boxed{33}$. (Correct final answer)
Reward Model	**Model correctly identifies incorrect reasoning**	Determine the value of x that satisfies $\sqrt[5]{x\sqrt{x^3}} = 3$.	We apply the power rule to $\sqrt[5]{x\sqrt{x^3}}$. (Incorrect reasoning) $$\sqrt[5]{x\sqrt{x^3}} = \sqrt[5]{x\sqrt{x^3}}^{\frac{1}{5}} = (x\sqrt{x^3})^{\frac{1}{5}} = (x^{(1/2)})^{(1/5)} = x^{\frac{1}{2} \times \frac{1}{5}} = x^{\frac{1}{10}}$$ (Incorrect reasoning) Substitute $x^{\frac{1}{10}}$ back into the original equation: $$3 = x^{\frac{1}{10}} \Rightarrow x^{\frac{1}{10}} = 3 \Rightarrow x^{(1/10)} = 3^2 \Rightarrow x = 3^2 \Rightarrow x = 9$$ Therefore, the value of x that satisfies $\sqrt[5]{x\sqrt{x^3}} = 3$ is $\boxed{9}$. (Correct final answer)

Table 8: Sample solutions generated by `Llama3.1-405B-Instruct` identified as having **incorrect** reasoning steps. We highlight in red incorrect judgements made by the models.

Strategy	Judgement correctness	Question	Solution
LLM-as-a-Judge	**Model correctly identifies incorrect reasoning**	Let x, y, and z be positive real numbers. Find the minimum value of $$\frac{(x^2+3x+1)(y^2+3y+1)(z^2+3z+1)}{xyz}.$$	Using AM-GM inequality, we have (Correct reasoning) $$x^2+1 \geq 2x \quad \text{and} \quad 3x+3 \geq 6\sqrt{x}$$ (Incorrect reasoning) $$(x^2+1)+(3x+3) \geq 2x+6\sqrt{x} \geq 2\sqrt{2x} \cdot 3\sqrt{x} = 6x$$ $$\Rightarrow x^2+3x+4 \geq 6x \Rightarrow \frac{x^2+3x+1}{x} \geq 5$$ (Incorrect reasoning) By similar process we get $\frac{y^2+3y+1}{y} \geq 5$ and $\frac{z^2+3z+1}{z} \geq 5$. Thus we can conclude that $$\frac{(x^2+3x+1)(y^2+3y+1)(z^2+3z+1)}{xyz} \geq 5 \times 5 \times 5 = 125$$ (Incorrect reasoning) Since $x = y = z = 1$ gives us $$\frac{(1^2+3(1)+1)(1^2+3(1)+1)(1^2+3(1)+1)}{1 \cdot 1 \cdot 1} = 5 \times 5 \times 5 = 125$$ and the minimum value of $$\frac{(x^2+3x+1)(y^2+3y+1)(z^2+3z+1)}{xyz}$$ is $\boxed{125}$.
LLM-as-a-Judge	**Model labels as incorrect, but the solution is correct**	A pen and its ink refill together cost $1.10. The pen costs $1 more than the ink refill. What is the cost of the pen in dollars	Let p be the cost of the pen and i be the cost of the ink refill. We know: 1. The total cost is $1.10 (Correct reasoning): $$p + i = 1.10$$ 2. The pen costs $1 more than the ink refill (Correct reasoning): $$p = i + 1$$ Solve for p: From the second equation, substitute $i = p - 1$ into the first equation: (Correct reasoning) $$p + (p - 1) = 1.10 \Rightarrow 2p - 1 = 1.10$$ $$\Rightarrow 2p = 2.10 \Rightarrow p = 2.10 : 2 = \boxed{1.05}$$
Reward Model	**Model correctly identifies incorrect reasoning**	There is a set of 1000 switches, each of which has four positions, called $A, B, C,$ and D. When the position of any switch changes, it is only from A to B, from B to C, from C to D, or from D to A. Initially each switch is in position A. The switches are labeled with the 1000 different integers $(2^x)(3^y)(5^z)$, where x, y, and z take on the values $0, 1, \ldots, 9$. At step i of a 1000-step process, the i-th switch is advanced one step, and so are all the other switches whose labels divide the label on the i-th switch. After step 1000 has been completed, how many switches will be in position A?	This is an extremely difficult problem to be solved by hand. (Incorrect reasoning) The number of switches in position A is $\boxed{650}$.
Reward Model	**Model labels as incorrect, but the solution is correct**	How many natural-number factors does N have if $N = 2^4 \cdot 3^3 \cdot 5^2 \cdot 7^2$?	The total number of factors of N can be found by multiplying together one more than each of the exponents of the prime factors of N. Since the prime factorization of N is: (Correct reasoning) $$N = 2^4 \cdot 3^3 \cdot 5^2 \cdot 7^2$$ the total number of factors is: (Correct reasoning) $$(4+1) \cdot (3+1) \cdot (2+1) \cdot (2+1) = 5 \cdot 4 \cdot 3 \cdot 3 = 180$$ So the answer is $\boxed{180}$.

Table 9: Comparison of SFT performance when selecting synthesized question-solution pairs with varying majority thresholds for determining whether to include the question in SFT data.

Min-votes	Data size	MATH Validation Accuracy
0	381K	**50.1**
8	339K	49.2
16	254K	44.4
24	160K	42.0

Table 10: Examples of paraphrases detected by our decontamination pipeline which will be missed by n-gram matching.

MATH Test Set Question	**Synthesized Question**
How many ordered triplets (a, b, c) of rational numbers are there where a, b, c are the roots of $x^3 + ax^2 + bx + c = 0$?	Find the number of ordered triplets (a, b, c) of real numbers such that the cubic equation $x^3 + ax^2 + bx + c = 0$ has roots $a, b,$ and c.
In how many ways can we seat 6 people around a round table if Fred and Gwen insist on sitting opposite each other? (Two seatings are considered equivalent if one is a rotation of the other.)	A circular table has 6 identical chairs placed around it. In how many ways can 6 people, including Alice and Bob, be seated around the table if Alice and Bob want to sit opposite each other? Two seating arrangements are considered the same if one is a rotation of the other.

C QUESTION-SOLUTION AUGMENTATION

C.1 MINIMUM MAJORITY VOTE ABLATION

To determine the answer to synthetically generated questions, we use majority voting as a proxy for ground truth answer. We conduct an ablation study to determine the threshold for a minimum number of majority votes. The questions for which the number of majority vote solutions is less than the threshold are removed. We generate 32 solutions per question for a small set of initial synthesized questions (after performing decontamination with MATH validation subset) and perform a comparison of varying the majority vote threshold from {0, 8, 16, 24}. Based on the results presented in Table 9, we select the threshold of 0 in our experiments.

C.2 CONTAMINATED EXAMPLES DETECTED BY LLMS

The decontamination pipeline described in Section 3.1 identifies questions that will be missed by a simple n-gram baseline. Using it we have effectively filtered out approximately 50K questions from the 569K newly synthesized questions, reducing the total from 569K to 519K.

We show two such examples in Table 10. Our dataset does have questions that are similar (but not equivalent) to MATH test set questions with sample pairs shown in Table 11.

Table 11: Examples of questions from OpenMathInstruct-2 which are similar (but not equivalent) to questions from the MATH test set.

MATH Test Set Question	Similar question from OpenMathInstruct-2
Determine the number of ways to arrange the letters of the word GAMMAS	Find the number of ways to arrange the letters of the word DETAIL
Factor $32x^3 - 4x^2 + 20x$	Factor the expression $x^6 - 20x^3 - 30$
Three points are chosen randomly and independently on a circle. What is the probability that all three pairwise distances between the points are less than the radius of the circle?	Three points are chosen uniformly at random on a circle. What is the probability that no two of these points form an obtuse triangle with the circle's center?
Compute $$\cos \frac{2\pi}{7} \cos \frac{4\pi}{7} \cos \frac{8\pi}{7}$$	Compute $$\cos \left(\frac{7\pi}{4} \right)$$
What is the remainder when 5^{30} is divided by 7?	What is the remainder when 5^{2005} is divided by 27?
What is the digit in the hundredths place of the decimal equivalent of $\frac{9}{160}$?	Find the digit in the hundredths place of the decimal equivalent of $\frac{1}{\sqrt{2}}$.

D LLM PROMPTS

D.1 SOLUTION AUGMENTATION PROMPT

Few-shot Prompt: Solution Augmentation

Here are some examples of questions and solutions followed by a new
  question that you need to solve.
Make sure to put the answer (and only answer) inside \boxed{{}}.

Question:
A parabola with equation $y=x^2+bx+c$ passes through the points $
(-1,-11)$ and $(3,17)$. What is c?

My solution:
From the question we know that points $(-1, -11)$ and $(3, 17)$ lie
 on the parabola. This means that when we substitute x and y
from these points into the equation $y = x^2 + bx + c$, the
equation must hold true. We substitute these two points into the
given equation to solve for c.

For the point $(-1, -11)$:

Substitute $x = -1$ and $ y = -11 $ into the equation:
\[-11 = (-1)^2 + b(-1) + c \Rightarrow -11 = 1 - b + c \Rightarrow
 -b + c = -12 \]

For the point $(3, 17)$:

Substitute $x = 3$ and $y = 17$ into the equation:
\[17 = (3)^2 + b(3) + c \Rightarrow 17 = 9 + 3b + c \Rightarrow 3b
 + c = 8 \]

In summary, we have the two equations
\begin{align*}
-b + c &= -12\\
3b + c &= 8
\end{align*}

To solve for c we can eliminate b by multiplying the first
equation by 3 and adding equations together.
Multiplying the first equation by 3, we have $3(-b + c) = 3 (-12) \
Rightarrow -3b + 3c = -36$. Adding equations together gives us
\[(-3b + 3c) + (3b + c) = -36 + 8 \Rightarrow -3b + 3b + 3c + c =
-28 \Rightarrow 4c = -28 \Rightarrow c = -28 : 4 \Rightarrow c = \
boxed{-7} \]

Question:
Let $f(x)$ be an odd function. Is $f(f(x))$ even, odd, or neither
?\n\nEnter "odd", "even", or "neither".

My solution:

To determine whether $f(f(x))$ is even, odd, or neither, we need to use the property of $f(x)$ being an odd function.

An odd function is defined as:
\[f(-x) = -f(x) \quad \text{for all } x \]

Given that $f(x)$ is odd, let's find $f(f(-x))$ and see how it relates to $f(f(x))$.

1. Substitute $-x$ into the function $f(x)$:
\[f(-x) \]

2. Since $f(x)$ is odd, apply the definition of an odd function:
\[f(-x) = -f(x) \]

3. Now substitute $-f(x)$ into the function f:
\[f(f(-x)) = f(-f(x)) \]

4. Again, using the fact that $f(x)$ is odd, apply the definition:
\[f(-f(x)) = -f(f(x)) \]

5. We have found that:
\[f(f(-x)) = -f(f(x)) \]

This matches the definition of an odd function.

So, the answer is:
\[\boxed{\text{odd}} \]

Question:
A rectangular box P is inscribed in a sphere of radius r. The surface area of P is 384, and the sum of the lengths of its 12 edges is 112. What is r?

My solution:
Let the dimensions of the rectangular box P be x, y, and z. We know the following:

1. The sum of the lengths of the edges of P is
\[4(x + y + z) = 112 \Rightarrow x + y + z = 112 : 4 \Rightarrow x + y + z = 28 \]

2. The surface area of P is
\[2xy + 2yz + 2xz = 384 \Rightarrow xy + yz + xz = 384 : 2 \Rightarrow xy + yz + xz = 192 \]

Since the box is inscribed in the sphere, the diagonal of the box is the diameter of the sphere. The length of the diagonal is $\sqrt{x^2 + y^2 + z^2}$

The diameter of the sphere is $2r$, so:
\[2r = \sqrt{x^2 + y^2 + z^2} \Rightarrow (2r)^2 = x^2 + y^2 + z^2 = (x + y + z)^2 - (2xy + 2yz + 2xz) \]

Substitute the known values:
\[4r^2 = 28^2 - 384 = 784 - 384 = 400 \Rightarrow r^2 = 100 \Rightarrow r = \boxed{10} \]

Question:
Let $\mathbf{a} = \begin{pmatrix} 2 \\ 1 \\ 5 \end{pmatrix}.$ Find the vector $\mathbf{b}$ such that $\mathbf{a} \cdot \mathbf{b} = 11$ and\n\[\mathbf{a} \times \mathbf{b} = \begin{pmatrix} -13 \\ -9 \\ 7 \end{pmatrix}.\]

My solution:
Let $\mathbf{b} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$.

First, use the dot product condition:
\[\mathbf{a} \cdot \mathbf{b} = 11 \Rightarrow 2x + y + 5z = 11 \]

Next, use the cross product condition:
\[\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 2 \\ 1 \\ 5 \end{pmatrix} \times \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -5y + z \\ 5x - 2z \\ -x + 2y \end{pmatrix} = \begin{pmatrix} -13 \\ -9 \\ 7 \end{pmatrix} \]

This gives us the system of equations:
\begin{align*}
2x + y + 5z = 11 \quad &(1) \\
-5y + z = -13 \quad &(2) \\
5x - 2z = -9 \quad &(3) \\
-x + 2y = 7 \quad &(4)
\end{align*}

Solve for x, y, and z step-by-step:

From (2), $z = 5y - 13$.
From (4), $x = 2y - 7$.

Substitute $z = 5y - 13$ into (1):
\[2(2y - 7) + y + 5(5y - 13) = 11 \Rightarrow 4y - 14 + y + 25y - 65 = 11 \Rightarrow 30y - 79 = 11 \Rightarrow 30y = 90 \Rightarrow y = 3 \]

Now find x and z:
\[x = 2y - 7 = 2(3) - 7 = -1 \]

\[z = 5y - 13 = 5(3) - 13 = 2 \]

Thus, the vector $\mathbf{b}$ is:
\[\mathbf{b} = \boxed{\begin{pmatrix} -1 \\ 3 \\ 2 \end{pmatrix}} \]

Question:
{question}

My solution:

D.2 QUESTION-SOLUTION AUGMENTATION PROMPTS

Few-shot prompt for GSM8K Question Augmentation

Help the user to create a new math problem similar to a given one.
Make the new problem reasonable and solvable.

Here are some examples of how to complete this task.

Problem:
Olivia has $23. She bought five bagels for $3 each. How much money does she have left?

Write another problem similar to this one:
Aiden has $35. He purchased eight pencils for $2 each and a notebook for $5. How much money does he have remaining?

Problem:
Michael had 58 golf balls. On tuesday, he lost 23 golf balls. On wednesday, he lost 2 more. How many golf balls did he have at the end of wednesday?

Write another problem similar to this one:
Sarah collected 72 seashells during her beach vacation. On Thursday, she gave 15 seashells to her friend as a souvenir. On Friday, she found 8 more seashells while exploring the shore. How many seashells did Sarah have at the end of Friday?

Problem:
Angelo and Melanie want to plan how many hours over the next week they should study together for their test next week. They have 2 chapters of their textbook to study and 4 worksheets to memorize. They figure out that they should dedicate 3 hours to each chapter of their textbook and 1.5 hours for each worksheet. If they plan to study no more than 4 hours each day, how many days should they plan to study total over the next week if they take a 10-minute break every hour, include 3 10-minute snack breaks each day, and 30 minutes for lunch each day?

Write another problem similar to this one:
Samantha and David are preparing for their upcoming science fair project. They have four different experiments to conduct and a research paper to write. Each experiment is estimated to take 2 hours, and the research paper will require 8 hours to complete. To stay focused and productive, they plan to take a 15-minute break for every 1.5 hours of work and have three 20-minute snack breaks each day. Additionally, they allocate 45 minutes for lunch each day

---

. If they want to limit their daily study time to 5 hours, how many days should they plan to work on their project over the next two weeks?

Problem:
Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?

Write another problem similar to this one:
Tom has 50 marbles, and his friend Jerry has 65 marbles. If they decide to play a game and bet 20 marbles each, how many marbles will they have left in total after the game?

Problem:
There were nine computers in the server room. Five more computers were installed each day, from monday to thursday. How many computers are now in the server room?

Write another problem similar to this one:
In a garden, there were 12 flowers. Every morning for a week (from Monday to Sunday), 3 more flowers were planted. How many flowers are there in the garden now?

Problem:
Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?

Write another problem similar to this one:
Sarah had 35 marbles. She gave some marbles to her friend Emma. Now Sarah has 18 marbles left. How many marbles did Sarah give to Emma?

Problem:
Sam bought a dozen boxes, each with 30 highlighter pens inside, for $10 each box. He rearranged five of these boxes into packages of six highlighters each and sold them for $3 per package. He sold the rest of the highlighters separately at the rate of three pens for $2. How much profit did he make in total, in dollars?

Write another problem similar to this one:
Amy purchased 8 crates, each containing 24 colorful markers, for $12 per crate. She decided to create sets of 4 markers each and

sell them for \$2 per set. The remaining markers she sold
individually at a rate of 5 markers for \$3. Calculate the total
profit Amy made, in dollars.

Problem:
There are 15 trees in the grove. Grove workers will plant trees in
the grove today. After they are done, there will be 21 trees. How
many trees did the grove workers plant today?',

Write another problem similar to this one:
In a garden, there are 25 rose bushes. The gardener plans to plant
some more rose bushes today. After planting, there will be a total
of 40 rose bushes in the garden. How many rose bushes will the
gardener plant today?

Here is the problem from the user:
{question}

Write another problem similar to this one. Start directly with the
problem statement and DO NOT include any phrases such as "Here is a
new problem similar to a given one". After the problem is
generated finish your response right away.

Few-shot Prompt 1: MATH Question Augmentation

Help the user to create a new math problem similar to a given one.
Make the new problem reasonable and solvable.

Here are some examples of how to complete this task.

Problem:
In the equation $$5x^2-kx+1=0$$ determine k such that the
difference of the roots be equal to unity.

Write another problem similar to this one:
Consider the quadratic equation: $$3x^2 + mx - 2 = 0$$
Find the value of m for which the sum of the roots is equal to 4.

Problem:
Solve the following equation
$\\ds\\f{3+x}{3x}=\\sqrt {\\ds\\f{1}{9}+\\ds\\f{1}{x}\\sqrt {\\ds\\f{4}{9}+\\ds\\f{2}{x^2}}}$

Write another problem similar to this one:

Solve the following equation:
$\frac{2-y}{4y} = \sqrt{\frac{1}{16} + \frac{1}{y}\sqrt{\frac{9}{16} + \frac{3}{y^2}}}$

Problem:
In an infinitely decreasing geometric progression the sum of all
the terms occupying odd places is equal to 36, and that of all the
terms at even places equals 12.
Find the progression.

Write another problem similar to this one:
In an infinitely decreasing geometric sequence, the sum of all
terms in
positions that are multiples of 3 is equal to 54, while the sum of
all remaining terms is 126.
Find the first term and common ratio of this geometric sequence.

Problem:
Two railway stations are at a distance of 96 km from each other.
One train covers this distance 40 minutes faster than does the
other. The speed of the first train is 12 km/h higher than that of
the second.
Determine the speed of both trains.

Write another problem similar to this one:
Two airports are located 450 miles apart. A commercial airliner
flies this route 30 minutes faster than a smaller private jet. The
speed of the commercial airliner is 75 mph greater than that of the
private jet. Calculate the speed of both aircraft.

Here is the problem from the user:
{question}

Write another problem similar to this one. Start directly with the
problem statement and DO NOT include any phrases such as "Here is a
new problem similar to a given one". After the problem is
generated finish your response right away.

Few-shot Prompt 2: MATH Question Augmentation

Help the user to create a new math problem inspired by a given one.
Make the new problem reasonable and solvable.

Here are some examples of how to complete this task.

Problem:
In the equation $$5x^2-kx+1=0$$ determine k such that the difference of the roots be equal to unity.

Write another problem inspired by this one:
The roots x_1 and x_2 of the equation $$x^2-3ax+a^2=0$$ are such that $x_1^2+x_2^2=1.75$. Determine a.

Problem:
Solve the following equation $\\ds\\f{3+x}{3x}=\\sqrt{\\ds\\f{1}{9}+\\ds\\f{1}{x}\\sqrt{\\ds\\f{4}{9}+\\ds\\f{2}{x^2}}}$

Write another problem inspired by this one:
Solve the following equation $\\sqrt{1+x\\sqrt{x^2+24}}=x+1$

Problem:
In an infinitely decreasing geometric progression the sum of all the terms occupying odd places is equal to 36, and that of all the terms at even places equals 12. Find the progression.

Write another problem inspired by this one:
The sum of the terms of an infinitely decreasing geometric progression is equal to 56, and the sum of the squared terms of the same progression is 448. Find the first term and the common ratio.

Problem:
Two railway stations are at a distance of 96 km from each other. One train covers this distance 40 minutes faster than does the other. The speed of the first train is 12 km/h higher than that of the second. Determine the speed of both trains.

Write another problem inspired by this one:
A student was asked to multiply 78 by a two-digit number in which the tens digit was three times as large as the units digit; by mistake, he interchanged the digits in the second factor and thus obtained a product smaller than the true product by 2808. What was the true product?

Here is the problem from the user:
{question}

Write another problem inspired by this one.
Don't just change the numbers and context, but try to create a
problem that requires another approach to solve.
Start directly with the problem statement and DO NOT include any
phrases such as "Here is a new problem inspired by a given one".
After the problem is generated finish your response right away.

D.3 LLM-AS-A-JUDGE PROMPTS TO DETECT LOW-QUALITY SOLUTIONS

LLM-as-a-Judge: Prompt 1

Below is a mathematical question, followed by a solution and the
expected answer.
Evaluate whether the solution correctly addresses the question and
produces the expected answer.
The solution might be flawed but still result in the correct final
answer.
If there are significant mistakes during intermediate steps, respond
 with \boxed{{No}} even if the final answer is correct.
Summarize your reasoning in one sentence, then respond with either \
boxed{{Yes}} or \boxed{{No}}.

YOUR TASK

Question: {question}
Solution: {output}
Expected_answer: {expected_answer}

LLM-as-a-Judge: Prompt 2

You are given a question, a proposed solution, and a reference
answer.
Your job is to evaluate the proposed solution by comparing it with
the reference answer. Focus on both the final answer and the
reasoning process.
Please remember, even if the final answer produced by the solution
is correct, if the process is flawed or incorrect, it should still
be considered a wrong answer.

Follow instructions below:

Instructions:

1. Review the question: Start by understanding the question
thoroughly. Ensure that you grasp what is being asked before
evaluating the solutions.

2. Analyze the proposed solution: Break down the proposed solution
into its component steps. Identify the logical reasoning and
methodology used to arrive at the final answer.

3. Compare with the reference answer: Look at the reference answer
and its reasoning process. Determine how it approaches the problem
and the correctness of its steps.

4. Identify errors or inconsistencies: Check if the proposed
solution has any logical flaws, incorrect assumptions, or deviations
 from standard practices, even if the final answer appears correct.

5. Evaluate the correctness of the process: Assess whether the
process used in the proposed solution is valid and aligns with the
logical approach of the reference answer.

6. Provide a detailed assessment: Explain in detail whether the
proposed solution is correct or incorrect. If the solution is
correct but the reasoning is flawed, explain why it should still be

---

considered wrong. Conversely, if the final answer is incorrect but the process was logical, explain what went wrong.

YOUR TASK

Question: {question}
Solution: {output}
Reference answer: {expected_answer}

Summarize your reasoning within 500 words, then respond with either \boxed{{Yes}} or \boxed{{No}}.
Remember to put only the final conclusion "Yes" or "No" in \boxed {{}}.

D.4 LLM-AS-A-JUDGE FOR DECONTAMINATION

LLM Prompt for Decontamination

I will now give you two questions Original question and Candidate question, please help me determine if the following two questions are the same.

Original question: {question}
Candidate question: {candidate}

Disregard the names and minor changes in word order that appear within.
If their question prompts are very similar and, without considering the solution process, they produce the same answer, we consider them to be the same question.
Please respond with only "True" or "False" based on your judgment.
Do not respond with anything else.

D.5 LLM-AS-A-JUDGE FOR EVALUATION

LLM Prompt for Final Evaluation

You will be asked to look at the two answers (predicted and expected) to a math problem and to judge whether they are equivalent within the context of the problem.

Please first explain your reasoning in a couple of sentences. Then respond with only Yes or No as your judgement on whether the two answers are the same.
When comparing answers only perform trivial simplifications.

Here are a few examples.

Example 1:
Problem: Factor $7x^3 - 21x^2 + 14x$.
Predicted answer: $7x(x - 2)(x - 1)$
Expected answer: $7x(x-1)(x-2)$

Reasoning: The order of the factors does not matter, so the answers are the same.
Judgement: Yes

Example 2:
Problem: A rectangle has a length of 6 meters and a width of 2 meters. If the length is reduced by 3 meters and the width is halved, what is the new area of the rectangle in square meters?
Predicted answer: 3/2
Expected answer: 1.5

Reasoning: 3/2 is the same as 1.5
Judgement: Yes

Example 3:
Problem: Simplify the expression $\sqrt{{7!}}$, where $n!$ stands for $n\cdot(n-1)\cdot(n-2)\cdots \cdot 2\cdot 1$.
Predicted answer: 71
Expected answer: 12\sqrt{{35}}.

Reasoning: This is non-trivial to simplify, so the answers are different.
Judgement: No

Example 4:
Problem: What is the simplified form of the expression $\sqrt{{98 x^{{3}} y^{{5}} z}}$?
\begin{{align*}}
\text{{A)}} & 2 x y z \sqrt{{7 x y z}} &
\text{{B)}} & 7 x^{{2}} y^{{2}} \sqrt{{2 y z}}
\\
\text{{C)}} & 7 x y^{{2}} \sqrt{{2 x y z}} &
\text{{D)}} &49 x y^{{2}} \sqrt{{2 x y z}}
\\
\end{{align*}}
Predicted answer: 7 x y^{{2}} \\sqrt{{2 x y z}}
Expected answer: C

Reasoning: Predicted answer is the same as the expected answer choice C.
Judgement: Yes

Example 5:
Problem: A line segment of length 5 has one endpoint at $(1, 2)$ and the other endpoint at $(4, b)$. Find all possible values of b, separated by commas.
Predicted answer: −2, 6
Expected answer: 6, −2

Reasoning: The order doesn't matter in the context of the problem.
Judgement: Yes

Example 6:
Problem: Solve $\tan x = \sin x$ for $0 \le x \le 2 \pi.$ Enter all the solutions, separated by commas.
Predicted answer: 0, \pi
Expected answer: 0,\pi,2\pi.

Reasoning: Number of solutions is different.
Judgement: No

YOUR TASK

Problem: {problem}
Predicted answer: {predicted_answer}
Expected answer: {expected_answer}