MAmmoTH2: Scaling Instructions from the Web

◇**Xiang Yue**[*], ♠**Tuney Zheng**[*], ♠**Ge Zhang**[*], ♠**Wenhu Chen**[*]

◇Carnegie Mellon University, ♠University of Waterloo

xyue2@andrew.cmu.edu   wenhuchen@uwaterloo.ca

https://tiger-ai-lab.github.io/MAmmoTH2/

Abstract

Instruction tuning improves the reasoning abilities of large language models (LLMs), with data quality and scalability being the crucial factors. Most instruction tuning data come from human crowd-sourcing or GPT-4 distillation. We propose a paradigm to efficiently harvest 10 million naturally existing instruction data from the pre-training web corpus to enhance LLM reasoning. Our approach involves (1) recalling relevant documents, (2) extracting instruction-response pairs, and (3) refining the extracted pairs using open-source LLMs. Fine-tuning base LLMs on this dataset, we build `MAmmoTH2` models, which significantly boost performance on reasoning benchmarks. Notably, `MAmmoTH2-7B`'s (Mistral) performance increases from 11% to 36.7% on MATH and from 36% to 68.4% on GSM8K without training on any in-domain data. Further training `MAmmoTH2` on public instruction tuning datasets yields `MAmmoTH2-Plus`, achieving state-of-the-art performance on several reasoning and chatbot benchmarks. Our work demonstrates how to harvest large-scale, high-quality instruction data without costly human annotation or GPT-4 distillation, providing a new paradigm for building better instruction tuning data.

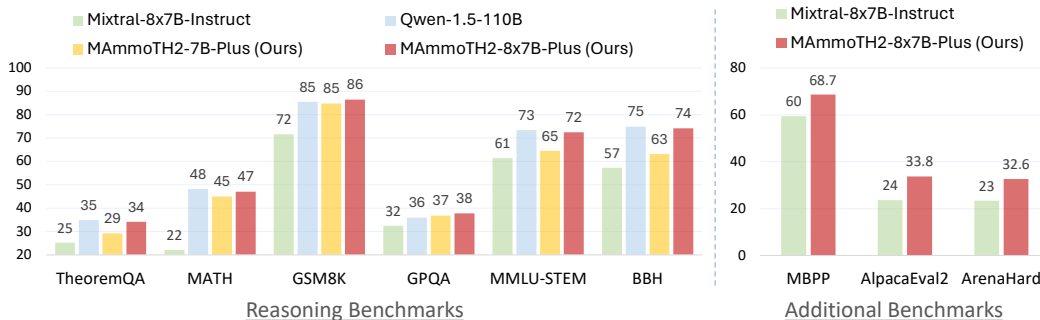



Figure 1: Overview of `MAmmoTH2-Plus` results. The `MAmmoTH2-8x7B-Plus` variant outperforms Mixtral-Instruct on reasoning benchmarks, matching Qwen-1.5-110B with only 13B active parameters. It also surpasses Mixtral-Instruct by around 10 points on general code and chatbot benchmarks.

1 Introduction

Reasoning is a fundamental aspect of human cognition and problem-solving [Clark et al., 2018, Hendrycks et al., 2021a, Cobbe et al., 2021, Rein et al., 2023, Yue et al., 2023a]. Proficiency in

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

---

[*]All of the authors are core contributors to the project.

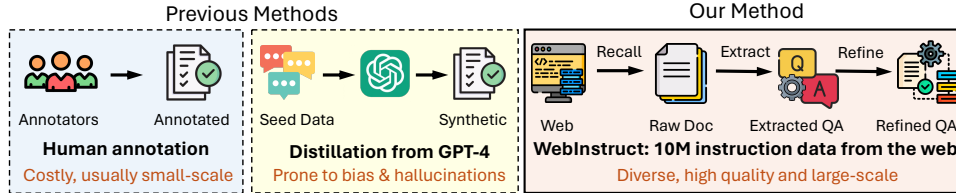



Figure 2: Comparison between our dataset curation method and previous studies.

reasoning is essential for advancing scientific knowledge, developing new technologies, and making informed decisions in various contexts. Recently, large language models (LLMs) [Brown et al., 2020, Ouyang et al., 2022, Touvron et al., 2023a,b, Achiam et al., 2023, Team et al., 2023] have shown remarkable progress in various NLP tasks. However, their ability to perform complex reasoning tasks [Lin et al., 2024] in the domains of mathematics, science, and engineering is still limited.

Recent studies have extensively explored how to enhance base LLMs' reasoning abilities. The two main approaches are continued training and instruction tuning. Continued training trains LLMs on large-scale filtered documents [Lewkowycz et al., 2022, Taylor et al., 2022, Azerbayev et al., 2023, Shao et al., 2024, Ying et al., 2024]. Instruction tuning seeks to employ supervised fine-tuning loss on, usually small-scale, high-quality instruction-response pairs [Ouyang et al., 2022, Chung et al., 2024]. While human-annotated instruction datasets [Cobbe et al., 2021, Hendrycks et al., 2021b, Amini et al., 2019] are often limited in scale, recent studies [Yu et al., 2023, Yue et al., 2023b, Toshniwal et al., 2024, Li et al., 2024a, Tang et al., 2024] attempt to prompt GPT-4 with seed data to increase the scalability. However, the synthesized instruction data becomes highly biased, not diverse, and prone to a high degree of hallucination.

To address these limitations, we propose to **discover naturally existing instruction data from the web** (Figure 2). We argue that the pre-training corpus (e.g., Common Crawl) already contains a vast amount of high-quality instruction data for LLM reasoning. For example, the web corpus contains a large amount of educational materials in the form of instruction-following pairs. These documents range across various domains like math, science, engineering, and humanities. Such readily available instruction data is not only diverse but also of high quality. However, such instruction data is highly dispersed across the corpus, which makes it particularly challenging to discover.

In this paper, we aim to mine these instruction-response pairs from the web using a three-step pipeline. **(1) Recall step**: We create a diverse seed dataset by crawling several quiz websites. We use this seed data to train a fastText model [Joulin et al., 2016] and employ it to recall documents from Common Crawl [Computer, 2023]. GPT-4 is used to trim down the recalled documents by their root URL. We obtain 18M documents through this step. **(2) Extract step**: We utilize open-source LLMs like Mixtral [Jiang et al., 2024] to extract Q-A pairs from these documents, producing roughly 5M candidate Q-A pairs. **(3) Refine step**: After extraction, we further employ Mixtral-8×7B [Jiang et al., 2024] and Qwen-72B [Bai et al., 2023] to refine [Zheng et al., 2024b] these candidate Q-A pairs. This refinement operation aims to remove unrelated content, fix formality, and add missing explanations to the candidate Q-A pairs. This refinement operation is pivotal to maintaining the quality of the mined Q-A pairs. Eventually, we harvest a total of 10M instruction-response pairs through these steps. Unlike existing instruction-tuning datasets, our dataset WEBINSTRUCT is purely mined from the Web without any human crowdsourcing or GPT-4 distillation.

We validate the effectiveness of WEBINSTRUCT by training MAmmoTH2 on various base models (Figure 1), including Mistral-7B [Jiang et al., 2023], Llama3-8B [Meta, 2024], Mixtral-8×7B [Jiang et al., 2024], and Yi-34B [Young et al., 2024]. MAmmoTH2 significantly outperforms the base models on seven held-out reasoning benchmarks: TheoremQA [Chen et al., 2023b], GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021b], ARC-C [Clark et al., 2018], MMLU-STEM [Hendrycks et al., 2021b], GPQA [Rein et al., 2023], and BBH [Suzgun et al., 2022]. MAmmoTH2-7B improves Mistral-7B's performance by an average of 14 absolute points, while MAmmoTH2-34B enhances Yi-34B's performance by an average of 5.8 absolute points. Notably, Mistral-7B's MATH accuracy can rise from 11.2% to 36.7% after training on WEBINSTRUCT. As our dataset contains no in-domain data from our evaluation benchmarks, this highlights the models' strong generalization ability.

We further enhance MAmmoTH2's performance on code generation, math reasoning, and instruction-following tasks by tuning it on open-source instruction datasets, including OpenHermes2.5 [Teknium,

2

2023], Code-Feedback [Zheng et al., 2024c], and Math-plus. The resulting model, `MAmmoTH2-Plus`, excels on seven reasoning benchmarks and other general tasks. `MAmmoTH2-7B-Plus` and `MAmmoTH2-8B-Plus` achieve state-of-the-art performance on TheoremQA, ARC-C, MMLU-STEM, GPQA, and BBH, and competitive results on MATH (45%) and GSM8K (85%). `MAmmoTH2-Plus` also performs well on general tasks, with `MAmmoTH2-7B-Plus` showing promising results on HumanEval and MBPP, and `MAmmoTH2-8×7B` leading the AlpacaEval 2.0 and Arena Hard leaderboards.

Interestingly, `MAmmoTH2-8B-Plus` and Llama-3-8B-Instruct, both tuned from Llama-3-base using datasets of the same size (10M), provide an apple-to-apple comparison. The only distinction is that Llama-3-8B-Instruct is trained on 10M human-annotated dataset while we do not require any human annotation. `MAmmoTH2-8B-Plus` outperforms Llama-3-Instruct by 6 points on reasoning tasks while matching its performance on general tasks, reflecting WEBINSTRUCT's cost-effectiveness advantage. `MAmmoTH2-Plus` consistently surpasses official instruction models like Mixtral-Instruct on chat benchmarks. These results demonstrate the effectiveness of our approach to scale up instruction data from the web and offer a new perspective for future instruction tuning studies.

2   WEBINSTRUCT

In this section, we outline the process of constructing WEBINSTRUCT. Specifically, we divide the data collection pipeline into three stages: (1) relevant document recall from the web corpus, (2) Q-A pair extraction from recalled document, and (3) Q-A pair refinement. The full pipeline is depicted in Figure 3 and an example for extraction and refinement is provided in Figure 4.

2.1   Recall from Common Crawl

In contrast to previous math-centric approaches [Paster et al., 2023, Wang et al., 2023c, Shao et al., 2024], we aim for broad coverage of disciplines such as math, science, engineering, etc. Therefore, careful balancing of the seed data is necessary to ensure diversity. However, publicly available training datasets are mostly limited to mathematics. To address this issue, we propose to crawl new

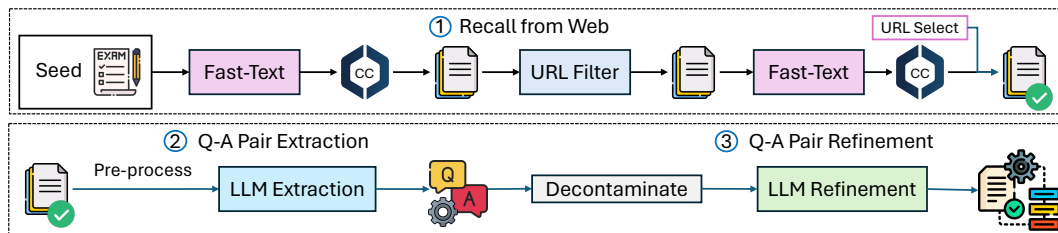



Figure 3: Step 1: Recall relevant documents from Common Crawl. Step 2: Extracting Q-A pairs. Step 3: Refine with the extracted Q-A pairs.

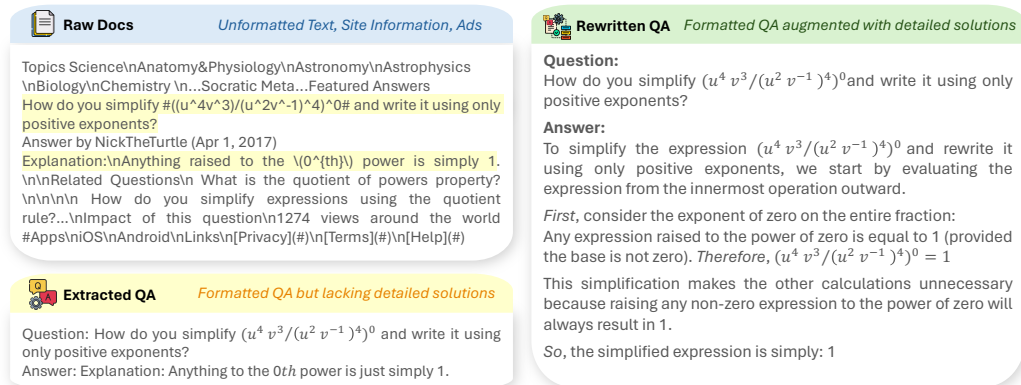



Figure 4: An illustrating example from WEBINSTRUCT for the extraction and refinement step.

3

exam problems from several educational websites. These sites contain diverse problems from various disciplines, helping to ensure diversity. We crawled 100K seed data as positive training examples and randomly selected 100K negative documents from CC [Computer, 2023] to train a fastText model [Joulin et al., 2016]. The trained fastText model is used to recall relevant documents. We employ the open-source fastText library with a vector dimension of 256 to train the model for 3 epochs, with a learning rate of 0.1, a maximum n-gram length of 3, and a maximum number of word occurrences of 3. We recalled 100B tokens using the trained fasttext model from an internal CC. These raw web documents are further grouped by their domains (root URL) and only domains with more than 1000 documents are retained. We extracted roughly 600K domains from the recalled documents. We then prompt GPT-3.5 to scan through the domains and automatically select those that might contain instruction data. Around 50K domains are further labeled as positive samples by GPT-3.5. Note that all the recalled documents in the first round are *not kept* for further usage in Q-A Pair Extraction and Refinement. Next, we sample documents from the selected domains as positive examples, and documents from the non-selected domains and general Common Crawl as negative examples to re-train an improved fastText classifier. The newly trained fastText classifier is used to recall documents. We recalled 40B tokens using the newly trained fastText model. We prompt GPT-4 to sift through the recalled domains again, ultimately leading to 18M raw documents, primarily originating from the desired websites.

2.2 Q-A Pair Extraction

We observe that a significant number of naturally existing Q-A pairs are present in the 18M documents. However, these Q-A pairs are interspersed with a high volume of noise such as ads, markups, boilerplate, etc. Our preliminary training on these raw documents only yields limited gains.

First, we carefully pre-process the HTML to pre-extract useful content from the recalled documents. This is mostly rule-based filtering to clean site information, ads, HTML boilerplate, etc. This step significantly reduces the document length for the next stage. We then prompt Qwen-72B [Bai et al., 2023] to identify the question and answer pairs from the preprocessed documents. Specifically, we provide a few in-context examples to help the model understand what to extract. We also allow the model to return void if no natural question-answer pairs exist. In this stage, only 30% of the recalled documents were identified as containing naturally existing Q-A pairs, resulting in roughly 5M Q-A pairs as our candidates for the next step. However, these candidates still contain a substantial amount of unrelated content and formality issues. Besides that, a large portion of the extracted Q-A pairs also lack explanations for how the answer is derived. Therefore, we propose to perform another round of refinement to increase the data quality.

To avoid contamination, we follow previous work [Shao et al., 2024] and filter out web pages containing questions or answers to all of our evaluation benchmarks. Specifically, we filter out all web pages that contain n-grams ($n = 10$) string matches with either the questions or answers.

2.3 Q-A Pair Refinement

To further improve the extracted Q-A pair candidates, we propose refining them using LLMs. In this step, we prompt Mixtral-22B×8 [Jiang et al., 2024] and Qwen-72B [Bai et al., 2023] to reformat the extracted Q-A pairs. If the answer does not contain any explanation, these two LLMs will attempt to complete the intermediate reasoning steps leading to the given answer. We adopt two models to increase the diversity of our dataset. Eventually, we harvest 10M Q-A pairs as our final instruction-tuning dataset WEBINSTRUCT.

2.4 Dataset Statistics

To better distinguish our dataset from the existing ones, we include a summarization table in Table 1. It can be observed that most SFT datasets contain less than 1M samples but are of high quality. XwinMath [Li et al., 2024a] is the largest dataset, scaling up to over 1M samples through GPT4 synthesis, while OpenMathInstruct [Toshniwal et al., 2024] has not been generated using GPT-4 but instead uses Mixtral-8x7B Jiang et al. [2024]. However, the seed data for both datasets is only based on GSM and MATH, leading to narrow domain coverage. In contrast, continue-training (CT) datasets are normally filtered from the web with much larger size, often exceeding 10B tokens and even rising to 120B tokens. However, continued pre-training on these massive datasets can be not only expensive

4

Table 1: The list of existing supervise-fine-tuning (SFT) and continue-training (CT) datasets. SFT datasets are primarily from academic NLP sources or synthesized by GPT-3.5/4 using seed data. CT datasets are larger but nosier. Our dataset falls between these two types.

Dataset	#Pairs	Domain	Format	Dataset Source
FLAN V2 [Chung et al., 2024]	100K	General	SFT	NLP data + Human CoT
Self-Instruct [Wang et al., 2023b]	82K	General	SFT	Generated by GPT3
GPT4-Alpaca [Taori et al., 2023]	52K	General	SFT	Generated by GPT4
SuperNI [Wang et al., 2022]	96K	General	SFT	NLP Datasets
Tora [Gou et al., 2023]	16K	Math	SFT	GSM+MATH Synthesis by GPT4
WizardMath [Luo et al., 2023]	96K	Math	SFT	GSM+MATH Synthesis by GPT4
MathInstruct [Yue et al., 2023b]	262K	Math	SFT	Math datasets Synthesis by GPT4
MetaMathQA [Yu et al., 2023]	395K	Math	SFT	GSM+MATH Synthesis by GPT3.5
XwinMath [Li et al., 2024a]	1.4M	Math	SFT	GSM+MATH Synthesis by GPT4
OpenMathInstruct [Toshniwal et al., 2024]	1.8M	Math	SFT	GSM+MATH Synthesis by Mixtral

Dataset	#Tokens	Domain	Format	Dataset Source
OpenWebMath [Paster et al., 2023]	12B	Math	LM	Filtered from Web
MathPile [Wang et al., 2023c]	10B	Math	LM	Filtered from Web
Cosmopeida [Ben Allal et al., 2024]	25B	General	LM	Synthesized by Mixtral
MINERVA [Lewkowycz et al., 2022]	38B	Math	LM	Filtered from Web
Proof-Pile-2 [Azerbayev et al., 2023]	55B	Math	LM	OpenWebMath+Arxiv+Code
Galactica [Taylor et al., 2022]	106B	Math & Sci.	LM	Filtered from Web
DeepseekMath [Shao et al., 2024]	120B	Math	LM	Recalled from Web
WEBINSTRUCT	(10M) 5B	Math & Sci.	SFT	Recall and Extracted from Web

but also ineffective due to the high noise ratio. WEBINSTRUCT, with roughly 5B tokens, strikes a good balance between scalability and quality. It approaches the scalability of common CT datasets while maintaining high quality through the three-step construction pipeline. This makes our dataset unique compared to other alternatives.

2.5 Additional Public Instruction Datasets

To further enhance the diversity and quality of our dataset, we fine-tune `MAmmoTH2` on several open-source instruction tuning datasets. These datasets are carefully selected based on their relevance to different reasoning subjects. Additionally, we consider some chat datasets to balance reasoning ability and general chat ability. The open-source datasets we incorporate are OpenHermes 2.5 [Teknium, 2023], Code-Feedback [Zheng et al., 2024c] and our Math-Plus, which is an augmented version of MetaMathQA (395K) [Yu et al., 2023] and Orca-Math (200K) [Mitra et al., 2024]. More details of the public datasets can be found in Appendix A.

3 Experimental Setup

3.1 Training Setup

We unify all the samples in our instruction dataset to conform to the structure of a multi-turn instruction tuning dataset. This standardization ensures that the fine-tuned models can process data consistently, regardless of the original dataset formats. We select the open-source models Mistral 7B [Jiang et al., 2023], Mixtral 8×7B [Jiang et al., 2024], Llama-3 8B [Meta, 2024], and Yi-34B [Young et al., 2024] as our base models. We fine-tune these models to validate our WEBINSTRUCT at multiple scales using the LLaMA-Factory [Zheng et al., 2024d] library. We use a learning rate of 5e-6 for Mistral 7B and 1e-5 for Mixtral, Llama-3 8B, and Yi 34B. The global batch size is set to 512 with a maximum sequence length of 4096. We employ a cosine scheduler with a 3% warm-up period for 2 epochs. To efficiently train the models, we utilize DeepSpeed [Rasley et al., 2020] with the ZeRO-3 stage. All the models are trained with 32 A100 GPUs.

3.2 Evaluation Datasets

To rigorously assess the capabilities of models in reasoning abilities across different domains, we utilize several widely used datasets, GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021b], TheoremQA [Chen et al., 2023b], BIG-Bench Hard (BBH) [Suzgun et al., 2022], ARC-C [Clark

5

Table 2: Main results on reasoning datasets. Models without the '-Instruct' suffix refer to the released base models. Results are taken from official papers or blogs when available; otherwise, we use our own evaluation script. Underscored results represent the best baseline scores under the size constraint. All models are inferred with few-shot CoT: TheoremQA (5-shot), MATH (4-shot), GSM8K (4-shot), GPQA (5-shot), MMLU-STEM (5-shot), BBH (3-shot), and ARC-C (8-shot).

Model	TheoremQA	MATH	GSM8K	GPQA	MMLU-ST	BBH	ARC-C	AVG
GPT-4-Turbo-0409	48.4	69.2	94.5	46.2	76.5	86.7	93.6	73.6
Parameter Size between 20B and 110B								
Qwen-1.5-110B	34.9	49.6	85.4	35.9	73.4	74.8	91.6	63.6
Qwen-1.5-72B	29.3	46.8	77.6	36.3	68.5	68.0	92.2	59.8
Deepseek-LM-67B	25.3	15.9	66.5	31.8	57.4	71.7	86.8	50.7
Yi-34B	23.2	15.9	67.9	29.7	62.6	66.4	89.5	50.7
Llemma-34B	21.1	25.0	71.9	29.2	54.7	48.4	69.5	45.7
Mixtral-8×7B	23.2	28.4	74.4	29.7	59.7	66.8	84.7	52.4
Mixtral-8×7B-Instruct	25.3	22.1	71.7	32.4	61.4	57.3	84.7	50.7
Intern-Math-20B	17.1	37.7	82.9	28.9	50.1	39.3	68.6	46.4
Trained only with WEBINSTRUCT *(All evaluations are held-out)*								
MAmmoTH2-34B	30.4	35.0	75.6	31.8	64.5	68.0	90.0	56.4
Δ over Yi	+7.2	+19.1	+7.7	+2.1	+2.9	+1.2	+0.5	+5.8
MAmmoTH2-8x7B	32.2	39.0	75.4	36.8	67.4	71.1	87.5	58.9
Δ over Mixtral	+9.2	+10.6	+1.0	+7.1	+7.4	+3.3	+2.8	+6.5
Continue trained with additional instruction datasets (All held-out except MATH and GSM8K)								
MAmmoTH2-8x7B-Plus	**34.1**	**47.0**	**86.4**	**37.8**	**72.4**	**74.1**	**88.4**	**62.9**
Δ over Qwen-1.5-110B	-0.8	-2.6	+1.0	+1.5	-1.0	-0.7	-4.0	-0.7
Parameter Size = 7B or 8B								
Deepseek-7B	15.7	6.4	17.4	25.7	43.1	42.8	47.8	28.4
Qwen-1.5-7B	14.2	13.3	54.1	26.7	45.4	45.2	75.6	39.2
Mistral-7B	19.2	11.2	36.2	24.7	50.1	55.7	74.2	38.8
Gemma-7B	21.5	24.3	46.4	25.7	53.3	57.4	72.5	43.0
Llemma-7B	17.2	18.0	36.4	23.2	45.2	44.9	50.5	33.6
WizardMath-7B-1.1	11.7	33.0	83.2	28.7	52.7	56.7	76.9	49.0
Abel-7B-002	19.3	29.5	83.2	30.3	29.7	32.7	72.5	42.5
Intern-Math-7B	13.2	34.6	78.1	22.7	41.1	48.1	59.8	42.5
Rho-1-Math-7B	21.0	31.0	66.9	29.2	53.1	57.7	72.7	47.3
Deepseek-Math-7B	25.3	34.0	64.2	29.2	56.4	59.5	67.8	48.0
Deepseek-Math-Instruct	23.7	44.3	82.9	31.8	59.3	55.4	70.1	52.5
Llama-3-8B	20.1	21.3	54.8	27.2	55.6	61.1	78.6	45.5
Llama-3-8B-Instruct	22.8	30.0	79.5	34.5	60.2	66.0	80.8	53.4
Trained only with WEBINSTRUCT *(All evaluations are held-out)*								
MAmmoTH2-7B	29.0	36.7	68.4	32.4	62.4	58.6	81.7	52.8
Δ over Mistral	+9.8	+25.5	+32.2	+7.7	+12.3	+2.9	+7.5	+14.0
MAmmoTH2-8B	32.2	35.8	70.4	35.2	64.2	62.1	82.2	54.3
Δ over Llama3	+12.2	+14.5	+15.6	+8.0	+8.6	+1.0	+3.6	+8.8
Continue trained with additional instruction datasets (All held-out except MATH and GSM8K)								
MAmmoTH2-7B-Plus	29.2	**45.0**	**84.7**	36.8	64.5	63.1	83.0	58.0
MAmmoTH2-8B-Plus	**32.5**	42.8	84.1	**37.3**	**65.7**	**67.8**	**83.4**	**59.1**
Δ over best baseline	+7.2	+0.7	+1.5	+2.8	+5.5	+1.8	+2.6	+5.7

et al., 2018], GPQA [Rein et al., 2023], MMLU-STEM [Hendrycks et al., 2021a]. These datasets collectively enable a comprehensive assessment of language models' reasoning prowess across a spectrum of complexity and realism. The details of the evaluation datasets are in Appendix B.

We further evaluate the models on additional code generation tasks (including HumanEval [Chen et al., 2021], MBPP [Austin et al., 2021] and their augmented version [Liu et al., 2024]), general LLM benchmarks like MMLU [Hendrycks et al., 2021a] and its recent robust and challenging version MMLU-Pro [TIGER-Lab, 2024]. We also consider chat benchmarks like MT-Bench [Zheng et al., 2024a], AlpacaEval 2.0 [Li et al., 2023b], and Arena Hard [Li et al., 2024b] to demonstrate the generalizability of WEBINSTRUCT and WEBINSTRUCT-PLUS on more general LLM benchmarks.

6

4 Main Results

4.1 Experimental Results on Reasoning Benchmarks

Table 2 presents our main results, with existing models partitioned into two tracks based on their parameter size. For 7B parameter models, we observe that our model trained solely with WEBIN-STRUCT achieves significant improvements over the base models. For instance, MAmmoTH2-7B boosts the performance of Mistral-7B by an average of 14 points. Notably, WEBINSTRUCT does not contain any training data from these evaluation benchmarks, making all evaluations essentially held-out. The substantial performance gains demonstrate the strong generalization capabilities of MAmmoTH2-7B. Similarly, MAmmoTH2-8B boosts the performance of Llama-3-8B-base by an average of 8.8 points. We also experiment with larger models like Yi-34B and Mixtral to show that the performance gains are consistent across the board. Notably, Yi-34B's performance on MATH also increases by 19% after training on WEBINSTRUCT.

Further tuning on several additional public datasets also significantly enhances performance. The MAmmoTH2-Plus model family achieves state-of-the-art results across the board. For example, MAmmoTH2-Plus's performance on TheoremQA, GPQA, and ARC-C represents the best-known results for any model under 10B parameters. MAmmoTH2-7B-Plus's performance on MATH and GSM is also close to the best-known results. We also show the results of the models solely trained on the additional public datasets in Appendix E.

An interesting comparison is between MAmmoTH2-8B-Plus and Llama3-Instruct, as both models are trained from the Llama3-base. Llama-3-instruct was trained on a 10M human-annotated instruction dataset along with public datasets, similar to WEBINSTRUCT combined with additional public datasets. Therefore, these two models are highly comparable. Our experiments show that MAmmoTH2-8B-Plus outperforms Llama3-Instruct by an average of 6% across the benchmarks. This substantial gain indicates that WEBINSTRUCT is highly cost-effective. For larger models, we found that MAmmoTH2-8x7B-Plus can even match the performance of Qwen-1.5-110B with only 13B active parameters. These results demonstrate the effectiveness of our scalable instruction tuning approach.

4.2 Additional Experimental Results

To further demonstrate the capabilities of our models beyond the reasoning benchmarks presented in Table 2, we conduct additional experiments to evaluate their performance on code generation, general language understanding, and instruction-following tasks. Table 3 showcases the results of various models on code generation tasks. The MAmmoTH2-7B-Plus model exhibits strong performance, achieving the highest average scores of 66.1 and 58.2 on HumanEval(+) and MBPP(+) datasets, respectively. It outperforms the official instruct counterparts like Mistral-7B-Instruct-v0.2 on these metrics, indicating its superior code generation abilities.

To assess the general language understanding and instruction-following capabilities of our models, we evaluate them on a range of benchmarks, as shown in Table 3. The MAmmoTH2-Plus models exhibit strong performance across these tasks, showcasing their versatility and robustness. For example, MAmmoTH2-8×7B-Plus achieves the highest scores on AlpacaEval 2.0 and Arena Hard leaderboards, surpassing competitive models like GPT-3.5-Turbo and Tulu-2-DPO-70B [Ivison et al., 2023].

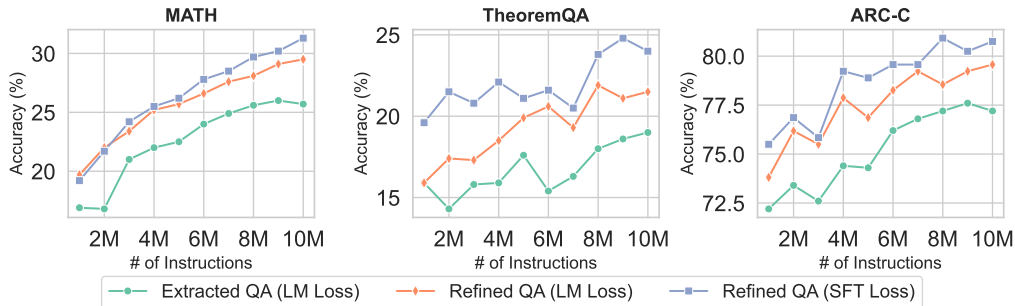



Figure 5: Mistral-7B model reasoning performance improves with scaling instructions. Additionally, SFT Loss is a more effective learning approach compared to LM Loss.

7

Table 3: Evaluation of code generation, instruction-following and MMLU(-Pro) performance for various models. We report the average of HumanEval(+) and MBPP (+) accuracy as the code generation performance (breakdown results are in Appendix D). Baseline scores are sourced from the original papers or the EvalPlus, MT-Bench, AlpacaEval 2.0, Arena Hard and MMLU-Pro leaderboards. ("-") indicates that the score is not available from the sources. `MAmmoTH2-Plus` exhibits strong general conversational ability and excels at multitask language understanding across a wide range of domains compared to their official instruct counterparts and larger models.

	Code Generation	MT-Bench	Alpaca Eval 2.0	Arena Hard	MMLU	MMLU-Pro
GPT-4-1106-preview	85.6 (77.5)	9.32	50.0	-	-	-
GPT-3.5-Turbo-1106	79.7 (70.2)	8.32	19.3	18.9	-	-
GPT-3.5-Turbo-0301	-	7.94	18.1	18.1	70.0	-
Tulu-2-DPO-70B	51.2 (43.0)	7.89	21.2	15.0	67.8	40.5
Llama-2-70b-chat	31.4 (26.5)	6.86	14.7	11.6	63.0	33.6
Yi-34B-Chat	38.7 (32.6)	7.86	27.2	23.1	73.5	42.1
Mistral-7B-Instruct-v0.2	43.4 (36.5)	7.60	17.1	12.6	60.8	30.8
Llama-3-8B-Instruct	<u>65.8</u> (<u>58.0</u>)	8.02	22.9	20.6	67.2	40.9
Mixtral-8×7B-Instruct-v0.1	52.3 (44.7)	**8.30**	<u>23.7</u>	<u>23.4</u>	**70.6**	41.0
`MAmmoTH2-7B-Plus`	**66.1** (**58.2**)	7.88	23.4	14.6	63.3	40.9
`MAmmoTH2-8B-Plus`	61.9 (53.3)	7.95	18.5	16.6	64.6	<u>43.4</u>
`MAmmoTH2-8x7B-Plus`	63.3 (55.3)	<u>8.20</u>	**33.8**	**32.6**	<u>68.3</u>	**50.4**

The strong performance of `MAmmoTH2` on code generation and general language understanding tasks, as evidenced by Table 3, demonstrates that our method does not overfit to the reasoning benchmarks. Instead, it shows the models' ability to generalize well to a wide range of tasks, highlighting their versatility and robustness. These additional experiments further validate the effectiveness of our WEBINSTRUCT in developing powerful and flexible language models.

5 Ablation Study

5.1 Scaling Effect of Instructions

We first investigate the impact of model scaling and loss functions on the performance of language models across three representative tasks: MATH, TheoremQA, and ARC-C. We train models with varying training samples (1M to 10M) using extracted QA and refined QA data, and compare the effectiveness of two training losses: LM loss and SFT loss. Figure 5 shows that increasing model size and using SFT Loss with synthetic data consistently improves accuracy across all tasks. These findings demonstrate the importance of model scaling and supervised fine-tuning with synthetic data for enhancing language model performance in various domains.

5.2 Comparison of Two Refined Models

To assess the effectiveness of the Q-A pair refinement process by different LLMs, we conducted experiments by training three Mistral-7B models: one on the data refined by Mixtral-22B×8, another on the data refined by Qwen-72B, and a third on the merged samples refined by both models. For a fair comparison, we trained the models with the same

Table 4: Comparison of the two data-refining LLMs. We train the three models with the same steps.

Data	GSM	MATH	MMLU-S	Theo.	ARC.
Mixtral	62.9	29.1	56.5	**26.1**	78.3
Qwen	65.4	28.9	**60.6**	23.5	80.8
Merged	**65.6**	**31.0**	60.5	24.8	**81.8**

9000 steps and a global batch size of 512. Our results show that the model trained on Mixtral-22B×8 refined data achieves comparable performance to the one trained on Qwen-72B refined data. The model trained on the merged samples consistently outperforms the models trained on data refined by individual LLMs. This demonstrates that combining data refined by different models mitigates potential biases that may arise from using a single model. By leveraging the strengths of both

8

Mixtral and Qwen, we obtain a more balanced and comprehensive dataset, preserving the diversity of naturally occurring web instructions while enhancing data quality. Unlike traditional distillation methods, our approach does not directly inherit knowledge from other models but instead utilizes LLMs for data extraction and refinement. This distinction helps preserve the original structure and content of instructions from diverse web sources, potentially reducing model-introduced biases.

5.3 Comparison of Different Domains and Sources.

To understand how each domain (e.g., math, science, others) and data source (e.g., forum websites and education websites) contribute to the training, we train Mistral 7B on the subsets of different domains and data sources. Details of how we obtain domain labels can be found in Appendix G.

As shown in Table 5, training on different domains and data sources leads to varied performance across the evaluation benchmarks. The education website data source consistently outperforms the forum data source, indicating the higher quality of educational questions. Interestingly, while the math domain excels on MATH, it does not lead to significant improvements on GSM8K, another math-focused dataset, suggesting that training on a single math dataset may not generalize well to other math benchmarks. Furthermore, training solely on the math domain does not yield substantial gains on science and STEM benchmarks, highlighting the need for diverse training. In contrast, the "Other" domain, which includes a diverse range of subjects, achieves the highest score on GSM8K, emphasizing the importance of diversity in the training data.

Table 5: Impact of different data domains and sources on model performance. All models are trained with identical steps; *Base* denotes the base model's performance.

Data Source	GSM	MATH	MMLU-S	Theo.	ARC.
Base	47.4	15.7	51.4	17.3	77.6
Forum	51.0	24.0	**54.7**	21.0	78.2
Education	**58.0**	**24.8**	54.3	**23.2**	**79.5**
Math	52.9	**27.3**	51.6	**21.7**	74.1
Science	54.4	23.7	**58.9**	21.0	**83.6**
Other	**59.4**	20.8	55.3	21.1	79.4

5.4 Effectiveness of Extract and Refine Steps

Our ablation studies highlight the essential role of the "Extract" and "Refine" steps in our data pipeline, demonstrating that these steps contribute significantly to model performance on a broad range of reasoning tasks.

Effectiveness of Extract and Refine Steps. To clarify the experimental settings used in our ablation study, we define each setup as follows:

- **(a) Recall**: Training on 18M recalled documents, serving as the baseline dataset.
- **(a') DeepSeek Math Recall** [Shao et al., 2024]: Deepseek Math 500B pre-training tokens.
- **(b) Extract**: Filtering the recalled documents to retain only high-quality question-answer pairs.
- **(c) Refine**: Further refining the extracted pairs by adding reasoning steps and clarifying responses.
- **(d) Public SFT**: Supervised fine-tuning using additional public fine-tuning data.
- **(d') DeepSeek Math SFT**: The SFT set from the DeepSeek Math.

The results, shown in Table 6, indicate that training solely on recalled documents with supervised fine-tuning, i.e., **Recall + Public SFT (a + d)**, achieves only modest gains, highlighting the importance of filtering and refining the initial corpus. Adding the "Extract" step, or **Recall + Extract + Public SFT (a + b + d)**, yields notable improvements by narrowing the training data to high-quality question-answer pairs, thus reducing token usage while preserving strong performance across benchmarks.

Incorporating the "Refine" step, or **Recall + Extract + Refine + Public SFT (a + b + c + d)**, further improves model accuracy by enhancing the rationale and clarity of responses, which is particularly beneficial for reasoning-intensive tasks. Additionally, comparisons with the **DeepSeek Math Corpus (a')** and **DeepSeek Math SFT (d')** demonstrate that while domain-specific data can produce high scores on targeted benchmarks, our generalized pipeline yields comparable or even superior results across a broader array of reasoning tasks.

9

Table 6: Summary of ablation study results using different training data. The **Base Model** is *DeepSeek Coder V1.5 7B* [Guo et al., 2024].

Setting	Model	#Tokens	TheorQA	MATH	GSM8K	MMLU-S	BBH	ARC-C	AVG
All evaluations are held-out									
Base Model	DS Coder v1.5 7B	-	18.3	22.3	47.9	47.0	53.5	62.4	41.9
+ (a)	-	28B	23.5	30.3	60.3	53.3	55.5	69.4	48.7
+ (a) + (b) + (c)	MAmmoTH2-DS	10B	**27.8**	33.8	64.0	**56.9**	58.5	**72.8**	**52.3**
+ (a')	DS Math Base	500B	25.3	**34.0**	**64.2**	56.4	**59.5**	67.8	51.2
All held-out except GSM and MATH									
+ (d)	-	2B	23.5	37.2	77.5	52.0	59.8	66.9	52.8
+ (a) + (d)	-	30B	27.2	39.2	79.2	55.6	60.3	71.5	55.5
+ (a) + (b) + (d)	-	9B	27.3	38.6	78.5	54.2	60.5	70.4	54.9
+ (a) + (b) + (c) + (d)	MAmmoTH2-DS-Plus	12B	**30.1**	43.8	80.1	**59.5**	**61.0**	**73.2**	**58.0**
+ (a') + (d')	DS Math Instruct	501B	23.7	**44.3**	**82.9**	59.3	55.4	70.1	56.0

In conclusion, these findings validate the necessity of both the "Extract" and "Refine" steps in building an effective instruction dataset from web-sourced data. This approach not only provides a scalable and economical alternative to human-annotated data but also enhances the model's generalization capability on a wide array of reasoning tasks.

5.5 Case Study

We further conduct a case study examining the quality of extracted and refined QA pairs from the dataset. We showcase some good and bad cases in Appendix J. We observe that the question/answer pairs extracted from well-formed exam and education websites are of high quality. The common issue is that a large portion of extracted answers do not contain intermediate rationale (chain-of-thought). This issue could lead to worse generalization.

Therefore, we prompt Mixtral and Qwen-72B to complete the intermediate steps. We observe that the success rate of such completion is relatively high. However, there are cases where the extracted question/answer pairs contain serious formatting issues, which pose challenges for the following refinement step. Besides these issues, we also observe that LLMs can sometimes modify the intention of the originally extracted content, causing hallucinations.

Figure 6: Quality distribution of 50 sampled refined QA examples.

To quantify the error percentages, we randomly sample 50 refined QA examples and ask the human annotators to compare whether the refined examples are correct and significantly better than the extracted ones in terms of format and intermediate solutions. As we can see from Figure 6, 78% examples have been improved after refinement and only 10% examples introduce hallucinations after refinement. Overall, our case study reveals that the harvested instruction tuning dataset is generally accurate with a low error rate.

6 Conclusion

In this paper, we argue that the web corpus contains a vast amount of high-quality instruction data across various domains. To mine this data, we develop a three-step pipeline consisting of recall, extraction, and refinement steps. Through this pipeline, we harvest WEBINSTRUCT, a total of 10M diverse, high-quality instruction-response pairs and train language models. Our experiments demonstrate that MAmmoTH2 exhibits significantly enhanced science reasoning abilities compared to the baseline models. Our work showcases the potential of harnessing the vast amount of instruction data in the web corpus to democratize the development of LLMs with enhanced reasoning capabilities.

10

References

J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *ArXiv preprint*, abs/2303.08774, 2023. URL `https://arxiv.org/abs/2303.08774`.

A. Amini, S. Gabriel, S. Lin, R. Koncel-Kedziorski, Y. Choi, and H. Hajishirzi. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, 2019. doi: 10.18653/v1/N19-1245. URL `https://aclanthology.org/N19-1245`.

J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le, et al. Program synthesis with large language models. *ArXiv preprint*, abs/2108.07732, 2021. URL `https://arxiv.org/abs/2108.07732`.

Z. Azerbayev, H. Schoelkopf, K. Paster, M. Dos Santos, S. M. McAleer, A. Q. Jiang, J. Deng, S. Biderman, and S. Welleck. Llemma: An open language model for mathematics. In *The Twelfth International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=4WnqRR915j`.

J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, et al. Qwen technical report. *ArXiv preprint*, abs/2309.16609, 2023. URL `https://arxiv.org/abs/2309.16609`.

L. Ben Allal, A. Lozhkov, G. Penedo, T. Wolf, and L. von Werra. Cosmopedia, 2024. URL `https://huggingface.co/datasets/HuggingFaceTB/cosmopedia`.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html`.

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. d. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, et al. Evaluating large language models trained on code. *ArXiv preprint*, abs/2107.03374, 2021. URL `https://arxiv.org/abs/2107.03374`.

W. Chen, X. Ma, X. Wang, and W. W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023a. URL `https://openreview.net/forum?id=YfZ4ZPt8zd`.

W. Chen, M. Yin, M. Ku, P. Lu, Y. Wan, X. Ma, J. Xu, X. Wang, and T. Xia. Theoremqa: A theorem-driven question answering dataset. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023b. URL `https://aclanthology.org/2023.emnlp-main.489/`.

H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024. URL `https://arxiv.org/pdf/2210.11416`.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv preprint*, abs/1803.05457, 2018. URL `https://arxiv.org/abs/1803.05457`.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al. Training verifiers to solve math word problems. *ArXiv preprint*, abs/2110.14168, 2021. URL `https://arxiv.org/abs/2110.14168`.

T. Computer. Redpajama: an open dataset for training large language models, 2023. URL `https://github.com/togethercomputer/RedPajama-Data`.

11

L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig. Pal: Program-aided language models. In *International Conference on Machine Learning*, pages 10764–10799. PMLR, 2023. URL https://arxiv.org/pdf/2211.10435.

Z. Gou, Z. Shao, Y. Gong, Y. Yang, M. Huang, N. Duan, W. Chen, et al. Tora: A tool-integrated reasoning agent for mathematical problem solving. *ArXiv preprint*, abs/2309.17452, 2023. URL https://arxiv.org/abs/2309.17452.

D. Guo, Q. Zhu, D. Yang, Z. Xie, K. Dong, W. Zhang, G. Chen, X. Bi, Y. Wu, Y. Li, et al. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*, 2024.

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the math dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021b. URL https://openreview.net/forum?id=7Bywt2mQsCe.

H. Ivison, Y. Wang, V. Pyatkin, N. Lambert, M. Peters, P. Dasigi, J. Jang, D. Wadden, N. A. Smith, I. Beltagy, et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *ArXiv preprint*, abs/2311.10702, 2023. URL https://arxiv.org/pdf/2311.10702.

A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al. Mistral 7b. *ArXiv preprint*, abs/2310.06825, 2023. URL https://arxiv.org/abs/2310.06825.

A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand, et al. Mixtral of experts. *ArXiv preprint*, abs/2401.04088, 2024. URL https://arxiv.org/abs/2401.04088.

A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov. Fasttext. zip: Compressing text classification models. *ArXiv preprint*, abs/1612.03651, 2016. URL https://arxiv.org/abs/1612.03651.

A. Lewkowycz, A. Andreassen, D. Dohan, E. Dyer, H. Michalewski, V. Ramasesh, A. Slone, C. Anil, I. Schlag, T. Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022. URL https://openreview.net/forum?id=IFXTZERXdM7.

C. Li, W. Wang, J. Hu, Y. Wei, N. Zheng, H. Hu, Z. Zhang, and H. Peng. Common 7b language models already possess strong math capabilities. *ArXiv preprint*, abs/2403.04706, 2024a. URL https://arxiv.org/abs/2403.04706.

T. Li, W.-L. Chiang, L. D. Evan Frick, B. Zhu, J. E. Gonzalez, and I. Stoica. From live data to high-quality benchmarks: The arena-hard pipeline, April 2024b. URL https://lmsys.org/blog/2024-04-19-arena-hard/.

X. Li, P. Yu, C. Zhou, T. Schick, O. Levy, L. Zettlemoyer, J. Weston, and M. Lewis. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*, 2023a.

X. Li, T. Zhang, Y. Dubois, R. Taori, I. Gulrajani, C. Guestrin, P. Liang, and T. B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 2023b.

Z. Lin, Z. Gou, T. Liang, R. Luo, H. Liu, and Y. Yang. Criticbench: Benchmarking llms for critique-correct reasoning. *ArXiv preprint*, abs/2402.14809, 2024. URL https://arxiv.org/abs/2402.14809.

12

J. Liu, C. S. Xia, Y. Wang, and L. Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36, 2024. URL `https://openreview.net/pdf?id=1qvx610Cu7`.

H. Luo, Q. Sun, C. Xu, P. Zhao, J. Lou, C. Tao, X. Geng, Q. Lin, S. Chen, and D. Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *ArXiv preprint*, abs/2308.09583, 2023. URL `https://arxiv.org/abs/2308.09583`.

Meta. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`, April 2024.

A. Mitra, H. Khanpour, C. Rosset, and A. Awadallah. Orca-math: Unlocking the potential of slms in grade school math. *ArXiv preprint*, abs/2402.14830, 2024. URL `https://arxiv.org/abs/2402.14830`.

M. Nye, A. J. Andreassen, G. Gur-Ari, H. Michalewski, J. Austin, D. Bieber, D. Dohan, A. Lewkowycz, M. Bosma, D. Luan, et al. Show your work: Scratchpads for intermediate computation with language models. In *Deep Learning for Code Workshop*, 2022. URL `https://openreview.net/forum?id=iedYJm92o0a`.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf`.

K. Paster, M. Dos Santos, Z. Azerbayev, and J. Ba. Openwebmath: An open dataset of high-quality mathematical web text. In *The Twelfth International Conference on Learning Representations*, 2023. URL `https://openreview.net/pdf?id=jKHmjlpViu`.

B. Peng, C. Li, P. He, M. Galley, and J. Gao. Instruction tuning with gpt-4. *ArXiv preprint*, abs/2304.03277, 2023. URL `https://arxiv.org/abs/2304.03277`.

J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3505–3506, 2020. URL `https://dl.acm.org/doi/10.1145/3394486.3406703`.

D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *ArXiv preprint*, abs/2311.12022, 2023. URL `https://arxiv.org/abs/2311.12022`.

V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, A. Raja, M. Dey, M. S. Bari, C. Xu, U. Thakker, S. S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey, R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Févry, J. A. Fries, R. Teehan, T. L. Scao, S. Biderman, L. Gao, T. Wolf, and A. M. Rush. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022. URL `https://openreview.net/forum?id=9Vrb9D0WI4`.

Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, M. Zhang, Y. Li, Y. Wu, and D. Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv preprint*, abs/2402.03300, 2024. URL `https://arxiv.org/abs/2402.03300`.

A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. URL `https://openreview.net/forum?id=uyTL5Bvosj`.

L. Sun, Y. Han, Z. Zhao, D. Ma, Z. Shen, B. Chen, L. Chen, and K. Yu. Scieval: A multi-level large language model evaluation benchmark for scientific research. *ArXiv preprint*, abs/2308.13149, 2023. URL `https://arxiv.org/abs/2308.13149`.

13

M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. V. Le, E. H. Chi, D. Zhou, , and J. Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. *ArXiv preprint*, abs/2210.09261, 2022. URL `https://arxiv.org/abs/2210.09261`.

Z. Tang, X. Zhang, B. Wan, and F. Wei. Mathscale: Scaling instruction tuning for mathematical reasoning. *ArXiv preprint*, abs/2403.02884, 2024. URL `https://arxiv.org/abs/2403.02884`.

R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic. Galactica: A large language model for science. *ArXiv preprint*, abs/2211.09085, 2022. URL `https://arxiv.org/abs/2211.09085`.

G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *ArXiv preprint*, abs/2312.11805, 2023. URL `https://arxiv.org/abs/2312.11805`.

Teknium. Openhermes 2.5: An open dataset of synthetic data for generalist llm assistants, 2023. URL `https://huggingface.co/datasets/teknium/OpenHermes-2.5`.

TIGER-Lab. Mmlu professional dataset. Hugging Face Dataset Hub, 2024. URL `https://huggingface.co/datasets/TIGER-Lab/MMLU-Pro`.

S. Toshniwal, I. Moshkov, S. Narenthiran, D. Gitman, F. Jia, and I. Gitman. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *ArXiv preprint*, abs/2402.10176, 2024. URL `https://arxiv.org/abs/2402.10176`.

H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *ArXiv preprint*, abs/2302.13971, 2023a. URL `https://arxiv.org/abs/2302.13971`.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288, 2023b. URL `https://arxiv.org/abs/2307.09288`.

X. Wang, Z. Hu, P. Lu, Y. Zhu, J. Zhang, S. Subramaniam, A. Loomba, S. Zhang, Y. Sun, and W. Wang. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. In *The 3rd Workshop on Mathematical Reasoning and AI at NeurIPS'23*, 2023a. URL `https://openreview.net/forum?id=u6jbcaCHqO`.

Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Naik, A. Ashok, A. S. Dhanasekaran, A. Arunkumar, D. Stap, E. Pathak, G. Karamanolakis, H. Lai, I. Purohit, I. Mondal, J. Anderson, K. Kuznia, K. Doshi, K. K. Pal, M. Patel, M. Moradshahi, M. Parmar, M. Purohit, N. Varshney, P. R. Kaza, P. Verma, R. S. Puri, R. Karia, S. Doshi, S. K. Sampat, S. Mishra, S. Reddy A, S. Patro, T. Dixit, and X. Shen. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, 2022. URL `https://aclanthology.org/2022.emnlp-main.340`.

Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, 2023b. URL `https://arxiv.org/abs/2212.10560`.

Z. Wang, R. Xia, and P. Liu. Generative ai for math: Part i–mathpile: A billion-token-scale pretraining corpus for math. *ArXiv preprint*, abs/2312.17120, 2023c. URL `https://arxiv.org/abs/2312.17120`.

J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022a. URL `https://openreview.net/forum?id=gEZrGCozdqR`.

14

J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022b. URL `https://openreview.net/pdf?id=_VjQlMeSB_J`.

C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, C. Tao, and D. Jiang. Wizardlm: Empowering large language models to follow complex instructions. *ArXiv preprint*, abs/2304.12244, 2023. URL `https://arxiv.org/abs/2304.12244`.

H. Ying, S. Zhang, L. Li, Z. Zhou, Y. Shao, Z. Fei, Y. Ma, J. Hong, K. Liu, Z. Wang, et al. Internlm-math: Open math large language models toward verifiable reasoning. *ArXiv preprint*, abs/2402.06332, 2024. URL `https://arxiv.org/abs/2402.06332`.

A. Young, B. Chen, C. Li, C. Huang, G. Zhang, G. Zhang, H. Li, J. Zhu, J. Chen, J. Chang, et al. Yi: Open foundation models by 01. ai. *ArXiv preprint*, abs/2403.04652, 2024. URL `https://arxiv.org/abs/2403.04652`.

L. Yu, W. Jiang, H. Shi, Y. Jincheng, Z. Liu, Y. Zhang, J. Kwok, Z. Li, A. Weller, and W. Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=N8N0hgNDRt`.

L. Yuan, G. Cui, H. Wang, N. Ding, X. Wang, J. Deng, B. Shan, H. Chen, R. Xie, Y. Lin, et al. Advancing llm reasoning generalists with preference trees. *ArXiv preprint*, abs/2404.02078, 2024. URL `https://arxiv.org/abs/2404.02078`.

X. Yue, Y. Ni, K. Zhang, T. Zheng, R. Liu, G. Zhang, S. Stevens, D. Jiang, W. Ren, Y. Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *ArXiv preprint*, abs/2311.16502, 2023a. URL `https://arxiv.org/abs/2311.16502`.

X. Yue, X. Qu, G. Zhang, Y. Fu, W. Huang, H. Sun, Y. Su, and W. Chen. Mammoth: Building math generalist models through hybrid instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2023b. URL `https://openreview.net/forum?id=yLClGs770I`.

Y. Zhang, Y. Luo, Y. Yuan, and A. C.-C. Yao. Automathtext: Autonomous data selection with language models for mathematical texts. *ArXiv preprint*, abs/2402.07625, 2024. URL `https://arxiv.org/abs/2402.07625`.

L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36, 2024a. URL `https://openreview.net/forum?id=uccHPGDlao`.

T. Zheng, S. Guo, X. Qu, J. Guo, W. Zhang, X. Du, C. Lin, W. Huang, W. Chen, J. Fu, et al. Kun: Answer polishment for chinese self-alignment with instruction back-translation. *ArXiv preprint*, abs/2401.06477, 2024b. URL `https://arxiv.org/abs/2401.06477`.

T. Zheng, G. Zhang, T. Shen, X. Liu, B. Y. Lin, J. Fu, W. Chen, and X. Yue. Opencodeinterpreter: Integrating code generation with execution and refinement. *ArXiv preprint*, abs/2402.14658, 2024c. URL `https://arxiv.org/abs/2402.14658`.

Y. Zheng, R. Zhang, J. Zhang, Y. Ye, Z. Luo, and Y. Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. *ArXiv preprint*, abs/2403.13372, 2024d. URL `https://arxiv.org/abs/2403.13372`.

K. Zhou, B. Zhang, J. Wang, Z. Chen, W. X. Zhao, J. Sha, Z. Sheng, S. Wang, and J.-R. Wen. Jiuzhang3.0: Efficiently improving mathematical reasoning by training small data synthesis models. *arXiv preprint arXiv:2405.14365*, 2024.

15

A   Details of Additional Public Instruction Tuning Datasets

- **OpenHermes 2.5** [Teknium, 2023]: The OpenHermes-2.5 dataset is a comprehensive collection of diverse data sources for instruction tuning, including 1M examples from datasets in math, science, and coding, alongside synthetic and chat-based data. It incorporates diverse sources such as Airoboros 2.2, CamelAI Domain Expert Datasets, ChatBot Arena, Collective Cognition, Evol Instruct, Glaive Code Assistant, GPT4-LLM, GPTeacher, Medical Tasks, MetaMath, SlimOrca, Platypus, ShareGPT, and Unnatural Instructions GPT4. We remove TheoremQA from Platypus as it is one of our test sets.

- **Code-Feedback** [Zheng et al., 2024c]: The Code-Feedback dataset is a multi-turn code generation and refinement dataset, containing 68,000 multi-turn interactions between users, code generation models, and compiler systems. It includes initial user instructions followed by compiler and user feedback. This dataset significantly enhances the model's multi-turn interaction coding ability.

- **Math-Plus**: This dataset combines public datasets such as MetaMathQA (395K) [Yu et al., 2023] and Orca-Math (200K) [Mitra et al., 2024]. Both of these datasets are generated by GPT-3.5/4 using GSM/MATH and other math datasets as the seed data. To further augment the dataset, we prompt GPT-4 to rewrite Q-A pairs from MATH training sets, adding an additional 300K examples to enhance the challenging problems. The total size of the Math-Plus dataset is 894K examples.

To ensure consistency and compatibility, we carefully align the format and structure of these additional datasets with WEBINSTRUCT. These supplementary datasets provide a rich resource for training models to answer questions and provide explanations across a wide range of topics, enhancing their versatility and applicability in real-world scenarios.

B   Details of Evaluation Datasets

To rigorously assess the capabilities of models in reasoning abilities across different domains, we utilize several widely used datasets. Each of these datasets is designed to challenge the models in different aspects of reasoning.

- **GSM8K** [Cobbe et al., 2021]: This test dataset contains 1.32K diverse grade school math problems, intended to test basic arithmetic and reasoning ability in an educational context.

- **MATH** [Hendrycks et al., 2021b]: Comprising 5000 intricate competition-level problems to evaluate the models' ability to perform complex mathematical reasoning.

- **TheoremQA** [Chen et al., 2023b]: Focused on applying mathematical theorems to solve advanced problems in fields such as mathematics, physics, and engineering, TheoremQA includes 800 questions that test the theoretical reasoning capabilities.

- **BIG-Bench Hard (BBH)** [Suzgun et al., 2022]: Consisting of 23 tasks previously found challenging for language models from BIG-Bench [Srivastava et al., 2023], BBH contains a total of 6511 challenging problems examining the capability of LLMs to solve them.

- **ARC-C** [Clark et al., 2018]: ARC includes questions derived from various grade-level science exams, testing models' ability to handle both straightforward and complex scientific queries. We use the challenge subset, which contains 1,172 test questions.

- **GPQA** [Rein et al., 2023]: This dataset provides "Google-proof" questions in biology, physics, and chemistry, designed to test deep domain expertise and reasoning under challenging conditions. We use the diamond subset containing 198 hard problems.

- **MMLU-STEM** [Hendrycks et al., 2021a]: Spanning 57 subjects across multiple disciplines, MMLU evaluates the breadth and depth of a model's knowledge in a manner akin to academic and professional testing environments. We select the STEM subset of MMLU with 3.13K problems.

These datasets collectively enable a comprehensive assessment of language models' reasoning prowess across a spectrum of complexity and realism. We further evaluate the models on additional code generation tasks (including HumanEval [Chen et al., 2021], MBPP [Austin et al., 2021] and their augmented version [Liu et al., 2024]), general LLM benchmarks like MMLU [Hendrycks et al., 2021a], and chat benchmarks like MT-Bench [Zheng et al., 2024a], AlpacaEval 2.0 [Li et al., 2023b], and Arena Hard [Li et al., 2024b] to demonstrate the generalizability of WEBINSTRUCT and WEBINSTRUCT-PLUS on more general LLM benchmarks.

16

C Related Work

Instruction Tuning. Instruction tuning is crucial for aligning large language models (LLMs) with end tasks. There are two main types of instruction tuning data: (1) human-written data, such as FLAN [Wei et al., 2022a], T0 [Sanh et al., 2022], and SuperNI [Wang et al., 2022], which assemble large instruction-tuning datasets from existing human-labeled datasets; and (2) synthesized data, like Self-Instruct [Wang et al., 2023b], WizardLM [Xu et al., 2023], and GPT4-Alpaca [Peng et al., 2023], which create instruction-tuning datasets by synthesizing from powerful LLMs like GPT-4 [Achiam et al., 2023]. Both types of instruction-tuning data have advantages and disadvantages. Human-written data is limited in size due to the high cost and in task diversity because existing human-labeled datasets mostly focus on a few NLP tasks. Although synthesized data can be generated at any scale, the high rate of hallucination can lead to significant quality degradation. Moreover, the diversity of synthesized data is heavily influenced by the seed data. Without a diverse seed dataset, the synthesized data will lack domain coverage.

Compared with Humpback [Li et al., 2023a], we focus on mining naturally existing instruction-response pairs from the web, rather than generating new instructions. This additional extraction step reduces corpus redundancy and enhances overall data quality.

Mathematics Reasoning. In recent years, there has been a growing interest in enhancing the mathematical reasoning abilities of large language models (LLMs). Three main approaches have been proposed to improve LLMs' mathematical reasoning skills:

- Prompting: Chain-of-thought-prompting (CoT) [Nye et al., 2022, Wei et al., 2022b] elicits LLMs' inherent reasoning ability by demonstrating intermediate reasoning steps. Program-of-thoughts-prompting (PoT) [Chen et al., 2023a, Gao et al., 2023] utilizes tools to further augment LLMs' math reasoning abilities. Subsequent work [Gou et al., 2023, Toshniwal et al., 2024, Yuan et al., 2024] combines CoT and PoT to maximize LLMs' reasoning ability.

- Continued Training: Enabling LLMs to solve mathematical problems has been a long-standing challenge. MINERVA [Lewkowycz et al., 2022] and Galactica [Taylor et al., 2022] were pioneers in continued training of LLMs to adapt to scientific domains for math and science reasoning. Open-source models like Llemma [Azerbayev et al., 2023], DeepSeek-Math [Shao et al., 2024], and Intern-Math [Ying et al., 2024] have surpassed MINERVA and Galactica on math benchmarks. These approaches mainly rely on using an efficient classifier to recall documents from Common Crawl to retrieve a massive high-quality math-related corpus (>100B tokens) to enhance math reasoning.

- Instruction Tuning: Instruction tuning aims to enhance LLMs' math reasoning skills by efficiently training on human-annotated public datasets like GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021b], and MathQA [Amini et al., 2019]. However, these datasets are often insufficient in size and diversity. Therefore, recent work [Yu et al., 2023, Yue et al., 2023b, Toshniwal et al., 2024, Luo et al., 2023, Li et al., 2024a] proposes augmenting them with strong commercial LLMs like GPT-4 [Achiam et al., 2023]. These methods can significantly boost LLMs' performance on in-domain math benchmarks but may fall short of generalization. Concurrently, Zhou et al. [2024] explored an efficient way to train a small LLM for math problem synthesis.

Our work combines continued training with instruction tuning to exploit the benefits of both approaches. Specifically, our dataset is recalled from Common Crawl like DeepSeekMath [Shao et al., 2024]. However, due to the significant level of noise in the raw corpus, we utilize a strong LLM to filter and clean the corpus to extract the instruction tuning pairs.

Science Reasoning. In addition to mathematical reasoning, there is growing interest in improving LLMs' general scientific reasoning ability in subjects like physics, biology, chemistry, computer science, etc. Several benchmarks, such as MMLU [Hendrycks et al., 2021a], TheoremQA [Chen et al., 2023b], Sci-Bench [Wang et al., 2023a], SciEval [Sun et al., 2023], and GPQA [Rein et al., 2023], have been developed to measure LLMs' reasoning ability on tasks beyond math. However, there has been less effort in curating high-quality training data for the science domain. Most datasets, like OpenWebMath [Paster et al., 2023], Proof-Pile [Azerbayev et al., 2023], and MathPile [Zhang et al., 2024], are heavily biased towards mathematics. In this work, we aim to generalize the pre-training data to broader subjects through our newly curated science seed data.

17

D Code Generation Results

We report the code generation results of our models and baselines in Table 7.

	HumanEval	HumanEval+	MBPP	MBPP+	Average	Average+
Mistral-7B	28.7	23.8	51.9	42.1	40.3	33.0
Gemma-7B	26.8	20.1	52.6	43.4	39.7	31.8
Llama-3-8B	33.5	29.3	61.4	51.6	47.5	40.5
Gemma-1.1-7B-Instruct	42.7	35.4	57.1	45.0	49.9	40.2
Mistral-7B-Instruct-v0.2	**75.0**	**70.1**	44.7	37.0	59.9	53.6
Llama-3-8B-Instruct	61.6	56.7	**70.1**	**59.3**	<u>65.9</u>	<u>58.0</u>
Mixtral-8×7B-Instruct-v0.1	45.1	39.6	59.5	49.7	52.3	44.7
MAmmoTH2- 7B-Plus	<u>72.1</u>	<u>65.9</u>	60.1	50.4	**66.1**	**58.2**
MAmmoTH2- 8B-Plus	63.4	57.9	60.4	48.6	61.9	53.3
MAmmoTH2- 8×7B-Plus	57.9	53.7	<u>68.7</u>	<u>56.9</u>	63.3	55.3

Table 7: Code generation results of different models. Baseline results are copied from the EvalPlus [Liu et al., 2024] leaderboard.

E Impact of Additional Public Instruction Tuning Datasets

In Appendix A, we introduce additional public instruction tuning datasets to further boost the model's reasoning performance. Here, we show the three setups of models trained on: 1) WEBINSTRUCT only; 2) Additional Public Datasets only; 3) WEBINSTRUCT + Additional Public Datasets (which we first trained on WEBINSTRUCT and then continued training on additional public datasets). The results are shown in Table 8.

Data	TheoremQA	MATH	GSM8K	GPQA	MMLU-ST	BBH	ARC-C	AVG
Mistral 7B Base								
WEBINSTRUCT	29.0	36.7	68.4	32.4	62.4	58.6	81.7	52.8
PUBLIC DATASETS	22.6	37.9	83.5	29.3	57.6	62.7	79.9	53.4
WEBINS.+PUBLIC.	**29.2**	**45.0**	**84.7**	**36.8**	**64.5**	**63.1**	**83.0**	**58.0**
Mixtral 8x7B Base								
WEBINSTRUCT	32.2	39.0	75.4	36.8	67.4	71.1	87.5	58.9
PUBLIC DATASETS	31.3	45.1	85.3	37.4	69.4	73.2	88.1	61.4
WEBINS.+PUBLIC.	**34.1**	**47.0**	**86.4**	**37.8**	**72.4**	**74.1**	**88.4**	**62.9**

Table 8: Impact of additional public instruction tuning datasets.

F Distributions of Website Domains in WEBINSTRUCT

Figure 7 show the distribution of the top websites in WEBINSTRUCT.

18

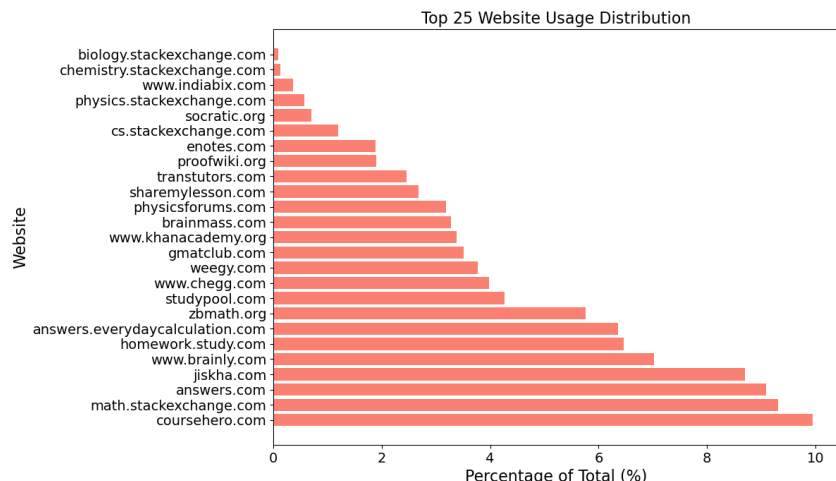



Figure 7: The distribution of the top websites in our instruction dataset.

G   Domain Distribution of WEBINSTRUCT

Figure 8 presents a breakdown of the WEBINSTRUCT by subject domains and data sources, providing insights into the composition and diversity of the mined instruction-response pairs. The subject labels are automatically annotated using the Llama-3-8B-Instruct model, while the distribution between education and forum data is obtained by analyzing the source URLs of the samples. The pie chart reveals that WEBINSTRUCT is predominantly composed of science-related subjects, with 81.69% of the data falling under the broad "Science" category. Within this category, Mathematics takes up the largest share at 68.36%, followed by Physics, Chemistry, and Biology. This highlights the dataset's strong emphasis on mathematical problem-solving and scientific reasoning. The remaining non-science categories, such as Business, Art & Design, and Health & Medicine, contribute to the diversity of the dataset. In terms of data sources, the vast majority (86.73%) of the instruction-response pairs come from exam-style questions, while forum discussions make up the remaining 13.27%. This source breakdown indicates that WEBINSTRUCT primarily consists of well-structured, educational content, supplemented by real-world discussions and inquiries from forums. The diverse subject coverage and the combination of education and forum data enable WEBINSTRUCT to capture a wide range of reasoning tasks and problem-solving scenarios.

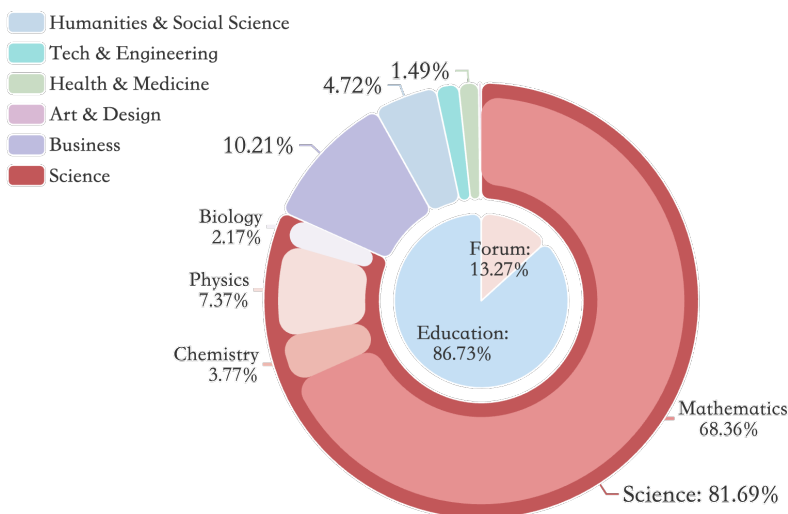



Figure 8: Breakdown of WEBINSTRUCT by subject domains and data sources.

19

H   Limitations of WEBINSTRUCT

Despite employing a three-step pipeline to ensure the quality of the mined instruction-response pairs, there may still be some noise and inaccuracies in the dataset, as mentioned in Figure 6. The extraction and refinement steps rely on the performance of the LLMs used, which may introduce biases and errors. Future work could explore more advanced techniques for data cleaning and validation, such as human-in-the-loop approaches or training a data selection model for filtering. Furthermore, while WEBINSTRUCT covers a wide range of subjects, including math, science, and engineering, there may be specific subdomains or advanced topics that are underrepresented, such as humanities and other daily chat topics. Expanding the coverage of the seed data to include broader and more diverse instruction-response pairs could further enhance the reasoning capabilities of LLMs in different areas.

I   Broader Social Impact

The development of MAmmoTH2 and the WEBINSTRUCT has the potential to positively impact education by providing the community with access to a large-scale, diverse set of instruction-response pairs across various subjects, particularly in mathematics and science. MAmmoTH2 can assist students in their learning process by offering step-by-step solutions, explanations, and targeted feedback, thus enhancing their understanding of complex concepts and problem-solving skills. Furthermore, the dataset and models can serve as valuable resources for educators to create more effective and engaging educational content, assignments, and assessments. However, it is important to consider potential limitations and biases in the dataset and the model. The quality and accuracy of the instruction-response pairs in WEBINSTRUCT may vary, and the model's outputs may reflect these inconsistencies. Therefore, it is crucial to employ rigorous data filtering and quality control measures to ensure the reliability and trustworthiness of the dataset and the model.

20

J Case Study

In this section, we aim to review some of the instances in WEBINSTRUCT. The data extracted from the website contains many issues. For example, the most common issue is that the answer is relatively short without any intermediate rationale, which could occur frequently from the quiz website with multi-choice questions. Another common issue is that the extraction destroys important formatting in the document, which leads to unrecoverable errors.

Good Case 1

Extracted:
`Question:` Find the distance from the point $\frac{3}{2}, \frac{3}{2}$ to the line $-\frac{37x}{10} - \frac{21y}{10} - \frac{7}{2} = 0$.

`Answer:` $61\sqrt{\frac{2}{905}}$.

Refined:
`Question:` Calculate the distance from the point $(\frac{3}{2}, \frac{3}{2})$ to the line defined by the equation $-\frac{37}{10}x - \frac{21}{10}y - \frac{7}{2} = 0$.

`Answer:` To find the distance 'd' from a point (x_0, y_0) to a line given by $Ax + By + C = 0$, we use the formula:
$$d = \frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}}.$$
For the point $\left(\frac{3}{2}, \frac{3}{2}\right)$ and the line $-\frac{37}{10}x - \frac{21}{10}y - \frac{7}{2} = 0$, substitute the coordinates into the formula:
$$d = \frac{\left|-\frac{37}{10} \cdot \frac{3}{2} - \frac{21}{10} \cdot \frac{3}{2} - \frac{7}{2}\right|}{\sqrt{\left(-\frac{37}{10}\right)^2 + \left(-\frac{21}{10}\right)^2}}.$$
Simplifying further, the exact simplified distance 'd' is $61\sqrt{\frac{2}{905}}$. Thus, the correct distance from the point to the line is $61\sqrt{\frac{2}{905}}$.

Figure A1: In this Refined version, the expression of both the question and the answer is clearer and more precise. It employs standardized mathematical symbols and terms, such as representing the coordinates of the point and the equation of the line as (x_0, y_0) and $Ax + By + C = 0$, respectively. Moreover, it explicitly provides the formula for calculating the distance. Through these revisions, readers can more easily understand how to compute the distance from a point to a line, making the solution process more transparent and easier to follow.

Good Case 2

Extracted:
`Question:` What are the two main ways that minerals are classified?
`Answer:` They can be classified by luster, streak, breakage, and hardness.

Refined:
`Question:` What are the primary criteria used for classifying minerals in mineralogy?
`Answer:` Minerals are primarily classified based on two main categories: physical properties and chemical properties. The physical properties include luster, streak, crystal form, hardness, density, and cleavage or breakage. Chemical properties involve the mineral's chemical composition and reactions with other substances. These characteristics help identify and distinguish different minerals in mineralogy.

Figure A2: Refining the extracted response provides a more comprehensive and accurate explanation of mineral classification in mineralogy. By expanding on the initial response to include both physical and chemical properties, the refined version offers a deeper understanding of how minerals are categorized and identified. This enriched content can enhance the model's ability to generate informative responses related to mineralogy.

21

Good Case 3

Extracted:
`Question:` How do you find theasymptotes for $Q(x) = (2x^2)/(x^2 - 5x - 6)$?
`Answer:` vertical asymptotes x=-1, x=6/n horizontal asymptote y=2 /n /n /n /n /n /n /n /n /n

Refined:
`Question:` How do you determine the vertical and horizontal asymptotes for the function $Q(x) = \frac{2x^2}{x^2-5x-6}$?
`Answer:` To find the vertical asymptotes, we need to identify the values of x that make the denominator equal to zero:

$$x^2 - 5x - 6 = (x - 6)(x + 1) = 0$$

This gives us the vertical asymptotes at $x = -1$ and $x = 6$.
For the horizontal asymptote, as x approaches infinity or negative infinity, the dominant terms in the numerator and denominator are both x^2. Thus, we compare their coefficients:
- Numerator's highest degree coefficient: 2 - Denominator's highest degree coefficient: 1
The horizontal asymptote is given by the ratio of these coefficients:

$$y = \frac{2}{1} = 2$$

Therefore, the vertical asymptotes are at $x = -1$ and $x = 6$, and the horizontal asymptote is $y = 2$.

Figure A3: The refined answer provides a more detailed explanation for finding the asymptotes. It includes the steps for solving for the vertical asymptotes by factoring the denominator and determining where it equals zero. It also explains how to find the horizontal asymptote by comparing the leading coefficients of the highest-degree terms in both the numerator and the denominator. The use of LaTeX formatting makes the mathematical notation clearer and easier to understand.

Good Case 4

Extracted:
`Question:` Which of the following is not an example of biased language? a. "You are too short to reach that." b. "Are you deaf?" c. "Move along, grandpa!" d. "I think I've been gypped.
`Answer:` D. I think I've been gypped.

Refined:
`Question:` Which of the following statements does not exhibit biased language?

    a. "You are too short to reach that."

    b. "Are you deaf?"

    c. "Move along, grandpa!"

    d. "I think I've been gypped."

`Answer:` The statement that does not exhibit biased language is:

    d. "I think I've been gypped."

Explanation: Biased language refers to words or phrases that stereotype, discriminate, or show prejudice towards a particular group. In this case, options a, b, and c contain biases towards people's physical abilities (height), hearing ability, and age, respectively. However, "I think I've been gypped" is an informal expression meaning to be cheated or deceived, which, while potentially offensive, does not directly target a specific group based on attributes like race, gender, or disabilities.

Figure A4: The refined version of the input presents the question and answers in a clearer, more structured format that improves readability and comprehension. Specifically, using a bulleted list for the options makes them distinct and easier to evaluate individually. Moreover, the addition of an explanation provides context and justification for the selected answer, enhancing the educational value of the content. This refined structure and the inclusion of an explanatory note make it a superior training example for models, as it helps in understanding not just the answer but the rationale behind it, promoting deeper learning and better pattern recognition in language processing tasks.

22

Good Case 5

Extracted:

`Question:` I am stucked on the following challenge: "If the line determined by two distinct points (x_1, y_1) and (x_2, y_2) is not vertical, and therefore has slope $(y_2 - y_1)/(x_2 - x_1)$, show that the point-slope form of its equation is the same regardless of which point is used as the given point." Okay, we can separate (x_0, y_0) from the form to get:

$$y(x_2 - x_1) - x(y_2 - y_1) = y_0(x_2 - x_1) - x_0(y_2 - y_1)$$

But how exclude this point (x_0, y_0) and leave only x, y, x_1, y_1, x_2, y_2 in the equation? UPDATE: There is a solution for this challenge:

$$(y_1 - y_2)x + (x_2 - x_1)y = x_2 y_1 - x_1 y_2$$

From the answer I found that

$$y_2(x - x_1) - y_1(x - x_2) = y(x_2 - x_1)$$

... but why this is true?

`Answer:` Thanks to saulspatz, the solution is to simply show that, whether we are using (x_1, y_1) or (x_2, y_2) as the given point, the equation does not change.

So both equations:

$$y - y_1 = m(x - x_1)$$
$$y - y_2 = m(x - x_2)$$

reduce to:

$$(y_1 - y_2)x + (x_2 - x_1)y = x_2 y_1 - x_1 y_2.$$

Refined:

`Question:` Demonstrate that the point-slope form of the equation of a line determined by two distinct points (x_1, y_1) and (x_2, y_2), where the line is not vertical, remains consistent regardless of which point is chosen as the reference. Why does the form of the equation not change when different reference points are used?

`Answer:` To address this challenge, we need to confirm that the point-slope equation of a line remains consistent regardless of the point chosen to derive it. Given two points (x_1, y_1) and (x_2, y_2) with a non-vertical line passing through them, the slope m of the line is $\frac{y_2 - y_1}{x_2 - x_1}$.

The point-slope form of a line's equation based on point (x_1, y_1) is:

$$y - y_1 = m(x - x_1)$$

Similarly, using point (x_2, y_2) gives:

$$y - y_2 = m(x - x_2)$$

These can be rewritten and compared to show their equivalence. Expanding both equations: - From the first equation: $y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$ - From the second equation: $y - y_2 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_2)$

We simplify these to find a common form. Multiplying both sides by $(x_2 - x_1)$ and rearranging terms, we find:

$$y_2(x - x_1) - y_1(x - x_2) = y(x_2 - x_1)$$

This confirms that both forms collapse to the same linear equation, showing the consistency of the point-slope form irrespective of the chosen reference point. Thus, it is demonstrated that the structural form of the equation remains unchanged, ensuring that the mathematical properties of the line—defined by its slope and points through which it passes—are invariant to the choice of the reference point used in its expression. This uniformity stems from the linearity and the constant slope defined by any two points on a straight line.

Figure A5: The original answer contains an irrelevant introductory sentence, which makes it less focused. In contrast, the refined version provides a clearer question by explicitly stating the problem of demonstrating the invariance of the point-slope form. The explanation in the answer is expanded to show step-by-step equivalence between the two forms and includes detailed reasoning about why the consistency holds. The added explanation reinforces the understanding of linear equations and the concept of constant slope, making the solution more comprehensive and pedagogically valuable.

23

Bad Case 1

Extracted:
`Question:` Sort $\left\{-3\pi, 9, \frac{17}{\sqrt{3}}, -4, \frac{47}{7}, \frac{3}{e}, \frac{3}{\sqrt{5}}\right\}$.

`Answer:` $\left\{-3\pi, -4, \frac{3}{e}, \frac{3}{\sqrt{5}}, \frac{47}{7}, 9, \frac{17}{\sqrt{3}}\right\}$.

Refined:
`Question:` Sort the following numbers in ascending order: $-3\pi, 9, \frac{17}{\sqrt{3}}, -4, \frac{47}{7}, \frac{3}{e}, \frac{3}{\sqrt{5}}$.

`Answer:` The sorted numbers in ascending order are: $-3\pi, -4, \frac{3}{e}, \frac{3}{\sqrt{5}}, \frac{47}{7}, 9, \frac{17}{\sqrt{3}}$.

Figure A6: The refined example does not enhance the clarity or quality of the information. The original formatting and question structure were already clear and concise, and the transformation into a slightly more verbose question-and-answer format does not add significant value. Additionally, both versions use proper LaTeX formatting for mathematical expressions, so there is no improvement in readability or technical accuracy. Essentially, the refinement introduces additional words without enhancing comprehension, which could potentially dilute the effectiveness of the model training by focusing on unnecessary stylistic alterations rather than content clarity or quality.

Bad Case 2

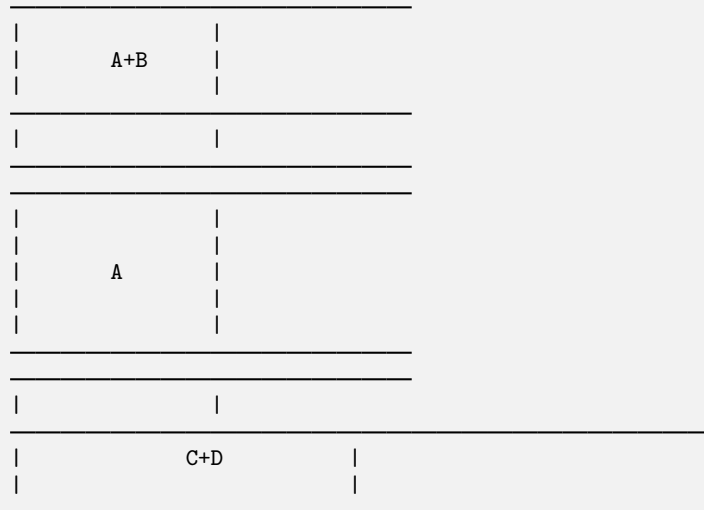
Extracted:
`Question:` Which curve (AD or SRAS) will be affected and which direction (leftward or rightward) will it shift when Congress goes to war against Canada?
`Answer:` When Congress falls into a war with Canada, the aggregate demand curve will be affected by moving to the left. Suppose Congress fights against Canada, then, it is likely that Congress will pass laws that advocate for increased tariffs on imports.
Spending on exports is an element of aggregate demand. Thus, its curve shifts rightwards, when its components, which are, spending on consumptions, government spending, or spending on investment increase.
On the other hand, Canada will not do business with the USA as usual when they fall out with Congress. For example, the expenses on exports to Canada will reduce.
Thus, the aggregate demand will fail to increase and shift on the left if there is less consumption from households, falls of investments from firms or when the demand for exports from a country like Canada falls.

Refined:
`Question:` How does a war between Congress and Canada affect the Aggregate Demand (AD) curve, and in which direction will it shift? `Answer:` When Congress declares war against Canada, the Aggregate Demand (AD) curve shifts to the left because Congress will implement increased tariffs on imports. Spending on exports, which is a component of aggregate demand, decreases due to these tariffs. The Short Run Aggregate Supply (SRAS) curve, however, is not directly affected by this situation. It mainly reflects changes in production costs and the availability of resources, which are not directly influenced by international conflicts unless they lead to disruptions in the domestic production process. In summary, the AD curve shifts left due to decreased exports and potentially lower consumer and business spending, while the SRAS curve remains relatively stable unless there are indirect impacts on domestic production.

Figure A7: The refined answer, while more structured and clearer in terms of the specific economic curves discussed, introduces inaccuracies that were not present in the original. Specifically, the refined version inaccurately states that spending on exports decreases due to tariffs, which contradicts the fundamental economic concept that tariffs affect imports rather than exports. Furthermore, the original answer provides a broader analysis of potential economic impacts, such as changes in household consumption and firm investment, which are omitted in the refined version. These omissions reduce the depth and accuracy of the economic analysis provided in response to the effects of war on economic curves.

24

Bad Case 3

Extracted:
`Question:` Is there a tool (LaTeX, JavaScript, Mathematica..) that allows one to draw Venn diagram automatically from subsets relations, e.g. $A \subset A+B, A \subset C, C \subset C+D, B \not\subset C$ would yield:

(Sorry for the pitiful Ascii drawing)

`Answer:` https://tex.stackexchange.com/questions/9681/how-to-draw-venn-diagrams-especially-complement-s-in-latex answers it in latex but I couldn't get their answer to work in my answer, as it used tikzpicture (seems unsupported here).

Refined:
`Question:` Is there any software for generating diagrams to show relationships among set members, for example, $A \subset A+B$?
`Answer:` One might try using LaTeX, which supports drawing with the TikZ package, although some features may not work in certain settings. For more details, check out this link: https://tex.stackexchange.com/questions/9681/how-to-draw-venn-diagrams-especially-complement-s-in-latex. If problems persist, updating the software or consulting the community might help.

Figure A8: The refined example's quality becomes bad because it omits most of the specific subset relations provided in the original query, making it unclear which relationships need to be visualized. The answer shifts the focus towards potential troubleshooting rather than directly addressing the user's need to generate diagrams based on specific subset conditions. Moreover, the original question included a sample ASCII diagram for clarity, which is entirely dropped in the refined version, removing a helpful virtual context.

25

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of WEBINSTRUCT in Appendix H. We acknowledge some noise and inaccuracies in the dataset due to the reliance on LLMs for extraction and refinement, which may introduce biases and errors. We also briefly discuss how to address these issues for future work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

26

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides extensive details on the experimental setups, including the datasets used, evaluation benchmarks, and the training details for the models. We describe the data collection pipeline, the training setup with specific parameters, and the evaluation methodology in a clear manner. This level of detail is sufficient to allow for the reproduction of the experimental results, thus supporting the main claims and conclusions of the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

27

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The paper submitted the data and code with the paper. We will also release the data and code to the public upon acceptance. Additionally, the paper includes detailed instructions for the experimental setup, dataset preparation, and model training processes.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The document provides detailed information about the training setup, including learning rate, global batch size, maximum sequence length, and the total number of GPUs employed for the models. It also mentions the use of a cosine scheduler for the learning rate adjustments and specifies the models trained with these settings.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

28

Justification: The paper provides error bars and statistical significance information for its experiments. Specifically, it includes detailed tables and figures that showcase the performance of various models with appropriate error bars. These visualizations are used to compare model performance across different benchmarks, such as reasoning and code generation tasks, ensuring that the reported results are statistically significant and accurately reflect the models' capabilities

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The paper provides detailed information about the computing resources used for the experiments. It specifies that all models were trained with 32 A100 GPUs and utilized DeepSpeed with the ZeRO-3 stage for efficient training. Additionally, the paper mentions the learning rates, global batch size, and maximum sequence length used during training. These details are sufficient for reproducing the experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The paper explicitly addresses ethical considerations in several sections. It ensures that the data used for training models is obtained from publicly available sources and does not include any private or sensitive information. The research follows guidelines

29

for responsible AI, including the use of diverse datasets to mitigate biases and a detailed assessment of model performance to ensure fairness and accuracy. Moreover, the methods and results are presented transparently, allowing for reproducibility.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper addresses both the potential positive and negative societal impacts of the research in Appendix I.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: The paper details several safeguards implemented to ensure the responsible release of the models and datasets. For instance, the dataset creation process includes filtering out web pages containing evaluation benchmark data to prevent contamination. Additionally, the extraction and refinement steps involve using strong language models to clean and enhance the quality of the data, reducing the risk of generating biased or low-quality outputs. These measures demonstrate a commitment to data integrity.

Guidelines:

- The answer NA means that the paper poses no such risks.

30

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper explicitly mentions and respects the licenses and terms of use for the various datasets and models utilized. It includes references to the original sources and acknowledges the contributions of existing assets, such as OpenWebMath, MathPile, and Cosmopeida. Additionally, the datasets used for fine-tuning, like OpenHermes 2.5 and Code-Feedback, are properly cited, ensuring that the terms of use are adhered to.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The paper introduces new assets, such as the WEBINSTRUCT dataset, which consists of 10 million instruction-response pairs harvested from the web. The documentation provided is thorough, detailing the steps for data collection, extraction, and refinement. Examples and case studies are included to illustrate the quality and structure of the data. Additionally, the datasets used for fine-tuning, such as OpenHermes 2.5 and Code-Feedback, are clearly listed and described, ensuring comprehensive documentation. Furthermore, the models developed in the paper, including the MAmmoTH2 and MAmmoTH2-Plus models, are planned to be open-sourced, which will include the necessary documentation and resources to facilitate their use by the research community.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.

31

- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper focuses on extracting and refining instruction-response pairs from existing web data and does not conduct new crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve research with human subjects. It primarily focuses on the development and evaluation of language models using data harvested from web sources, without any indication of human participant involvement or potential risks related to human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

32