

Math LLM Survey

Overview

Fine-tune models

序号	模型/方法	解决的问题	相关
			Dataset
1	Symbol-LLM	LLM在 符号能力任务 上性能不足	通过 3种符号数据收集方法 ,收集34到符号生成任务的数据
2	MAMMOTH	开源大模型和闭源大模型在 数学推理能力 上存在明显差距	编制了一个 指令调优数据集MathIn 1. 覆盖不同的数学领域和复杂程度 2. 将CoT和PoT进行集成
3	WizardMath	开源模型难以解决 复杂多步定量推理	通过调整后的Evol-Instruct方法生成 数学指令数据(集)
4	MetaMath	大语言模型在 数学推理 存在“逆转诅咒”现象	基于 GSM8K 和 MATH，执行 三种题引导 。结合 答案增强 ，提出了 MetaMathQA 数据集
5	Arithmo-Mistral-7B	/	模型训练数据集： 组合 MetaMath 训练拆分）、lila OOD（训练、验证拆分）和 MathInstruct（训练拆分）
6	Abel	Only SFT 也能train出很好的结果	without tools without continuing pretraining without reward model without RLHF ONLY using SFT can establish a new state-of-the performance across open-source on the GSM8k and MATH benchmark

RFT

序号	模型/方法	解决的问题	
			Pre-trained
7	RFT	使用SFT时，随着监督数据集的增加，模型性能提升较少	/

Datasets

序号	模型/方法	解决的问题	
8	MathPile	缺乏专门的为数学定制的语料库	Math-centric：专门为数学定制的 Diversity：超越了网页，整合了高 域的学术论文，以及来自StackExc High-Quality：进行了广泛的预处 Data Documentation：仔细记录

Tasks and Datasets

Tasks

1. Math Word Problem Solving

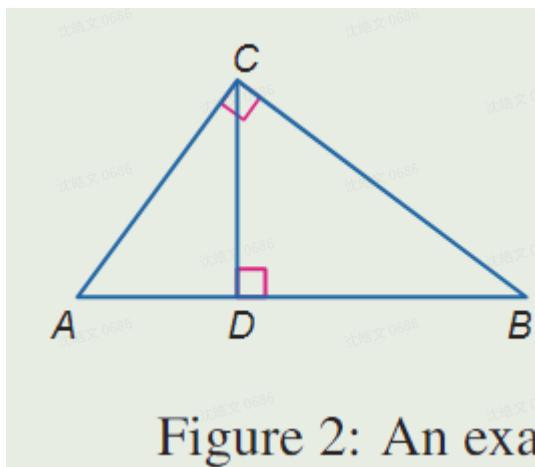
- a. Fundamentally, MWP solving can be seen as a complex relationship extraction problem, that is, the task of determining the complex relationships between numerical values in a given problem text
- b. e.g.,

Question: Bod has 2 apples and David has 5 apples. How many apples do they have in total?
Rationale: $x = 2 + 5$
Solution: 7

Table 1: A typical math word problem.

- 2. Theorem Proving: The problem is to prove the truth of a mathematical proposition (theorem) through a series of logical arguments (proofs).

3. Geometry Problem Solving



Question: In triangle ABC , $AD = 3$ and $BD = 14$. Find CD .

Choices: (A) 6.0 (B) 6.5 (C) 7.0 (D) 8.5

Answer: (B) 6.5

Figure 2: An example of geometry problems.

4. Math Question Answering: Questions and Answers Centered on Mathematical Reasoning

- a. e.g., Answer questions such as "Which kicker kicked the most field goals?" over the content of paragraphs.

Datasets

1. A summarization of mathematical reasoning datasets.

Dataset	Task	Size	Input	Output	Rationale	Domain
Verb395 (2014)	MWP	395	Question	Number	Equation	Math
Alg514 (2014)	MWP	514	Question	Number	Equation	Math
IL (2015)	MWP	-	Question	Number	Equation	Math
SingleEQ (2015)	MWP	508	Question	Number	Equation	Math
DRAW (2015)	MWP	1,000	Question	Number	Equation	Math
Dolphin1878 (2015)	MWP	1,878	Question	Number	Equation	Math
Dolphin18K (2016)	MWP	18,460	Question	Number	Equation	Math
MAWPS (2016)	MWP	3,320	Question	Number	Equation	Math
AIArith (2017)	MWP	831	Question	Number	Equation	Math
DRAW-1K (2017)	MWP	1,000	Question	Number	Equation	Math
Math23K (2017)	MWP	23,162	Question	Number	Equation	Math
AQuA (2017)	MWP	100,000	Question	Option	Natural language	Math
Aggregate (2018)	MWP	1,492	Question	Number	Equation	Math
MathQA (2019)	MWP	37,297	Question	Number	Program	Math
ASDiv (2020)	MWP	2,305	Question	Number	Equation	Math
HMWP (2020)	MWP	5,470	Question	Number	Equation	Math
Ape210K (2020)	MWP	210,488	Question	Number	Equation	Math
SVAMP (2021)	MWP	1,000	Question	Number	Equation	Math
GSM8K (2021)	MWP	8,792	Question	Number	Natural language	Math
IconQA (2021b)	MWP	107,439	Figure+Question	Option+Text span	X	Math
MathQA-Python (2021)	MWP	23,914	Question	Number	Python program	Math
ArMATH (2022)	MWP	6,000	Question	Number	Equation	Math
TabMWP (2022b)	MWP	38,431	Table+Question	Option+Number	Natural language	Math
MML (2015)	TP	57,882	Statement	Proof steps	X	Math
HolStep (2017)	TP	2,209,076	Statement	Proof steps	X	Math
Gamepad (2019)	TP	-	Statement	Proof steps	X	Math
CoqGym (2019)	TP	71,000	Statement	Proof steps	X	Math
HOList (2019)	TP	29,462	Statement	Proof steps	X	Math
IsarStep (2021)	TP	860,000	Statement	Proof steps	X	Math
PISA (2021)	TP	183,000	Statement	Proof steps	X	Math
INT (2021c)	TP	-	Statement	Proof steps	X	Math
NaturalProofs (2021)	TP	32,000	Statement	Proof steps	X	Math
NaturalProofs-Gen (2022a)	TP	14,500	Statement	Proof steps	X	Math
miniF2F (2022)	TP	488	Statement	Proof steps	X	Math
miniF2F+informal (2022a)	TP	488	Statement	Proof steps	X	Math
LeanStep (2022)	TP	21,606,000	Statement	Proof steps	X	Math
GEOS (2015)	GPS	186	Figure+Question	Option	X	Geometry
GeoShader (2017)	GPS	102	Figure+Question	Number	X	Geometry
GEOS++ (2017)	GPS	1,406	Figure+Question	Number	X	Geometry
GEOS-OS (2017)	GPS	2,235	Figure+Question	Option	Demonstration	Geometry
Geometry3K (2021a)	GPS	3,002	Figure+Question	Option	Logical form	Geometry
GeoQA (2021a)	GPS	4,998	Figure+Question	Option	Program	Geometry
GeoQA+ (2022)	GPS	12,054	Figure+Question	Option	Program	Geometry
UniGeo (2022a)	GPS/TP	14,541	Figure+Question	Option	Program	Geometry
Quarel (2019)	MathQA	2,771	Question	Option	Logical form	Math
McTaco (2019)	MathQA	13,225	Text+Question	Option	X	Time
DROP (2019)	MathQA	96,567	Passage+Question	Number+Text span	X	Math
Mathematics (2020)	MathQA	2,010,000	Question	Free-form	Number	Math
FinQA (2021c)	MathQA	8,281	Text+Table+Q	Number	Program	Finance
Fermi (2021)	MathQA	11,000	Question	Number	Program+Fact	Math
MATH (2021b)	MathQA	12,500	Question	Number	Natural language	Math
TAT-QA (2021)	MathQA	16,552	Text+Table+Q	Number+Text span	X	Finance
AMPS (2021b)	MathQA	5,000,000	Question	-	LaTeX	Math
MultiHierTT (2022)	MathQA	10,440	Text+Table+Q	Number+Text span	Expression	Finance
NumGLUE (2022b)	MathQA	101,835	Text+Question	Number+Text span	X	Math
Lila (2022a)	MathQA	134,000	Text+Question	Free-form	Python program	Math
FigureQA (2018)	VQA	1,000,000+	Figure+Question	Binary	X	Math
DVQA (2018)	VQA	3,487,194	Figure+Question	Text span	Number+Text span	Math
DREAM (2019)	ConvQA	10,197	Dialog+Question	Option	X	Math
EQUATE (2019)	NLI	-	Premise+Hypothesis	Binary	X	Math
NumerSense (2020)	Filling	13,600	Masked question	Word	X	Math
MNS (2020c)	IQ Test	-	Figure	Number	X	Math
P3 (2021)	Puzzle	397	Text	Program	X	Math
NOAHQA (2021)	ConvQA	21,347	Dialog+Question	Text span	Reasoning graph	Math
ConvFinQA (2022c)	ConvQA	3,892	Report+Dialog+Q	Number	Expression	Math
PGDP5K (2022)	Parsing	5,000	Figure+Question	Number	X	Geometry
GeoRE (2022a)	Parsing	12,901	Figure+Question	Number	X	Geometry
ScienceQA (2022a)	VQA	21,208	Context+Question	Option	Natural language	Science

Table 7: A summarization of mathematical reasoning datasets.

Detailed interpretation

Symbol-LLM

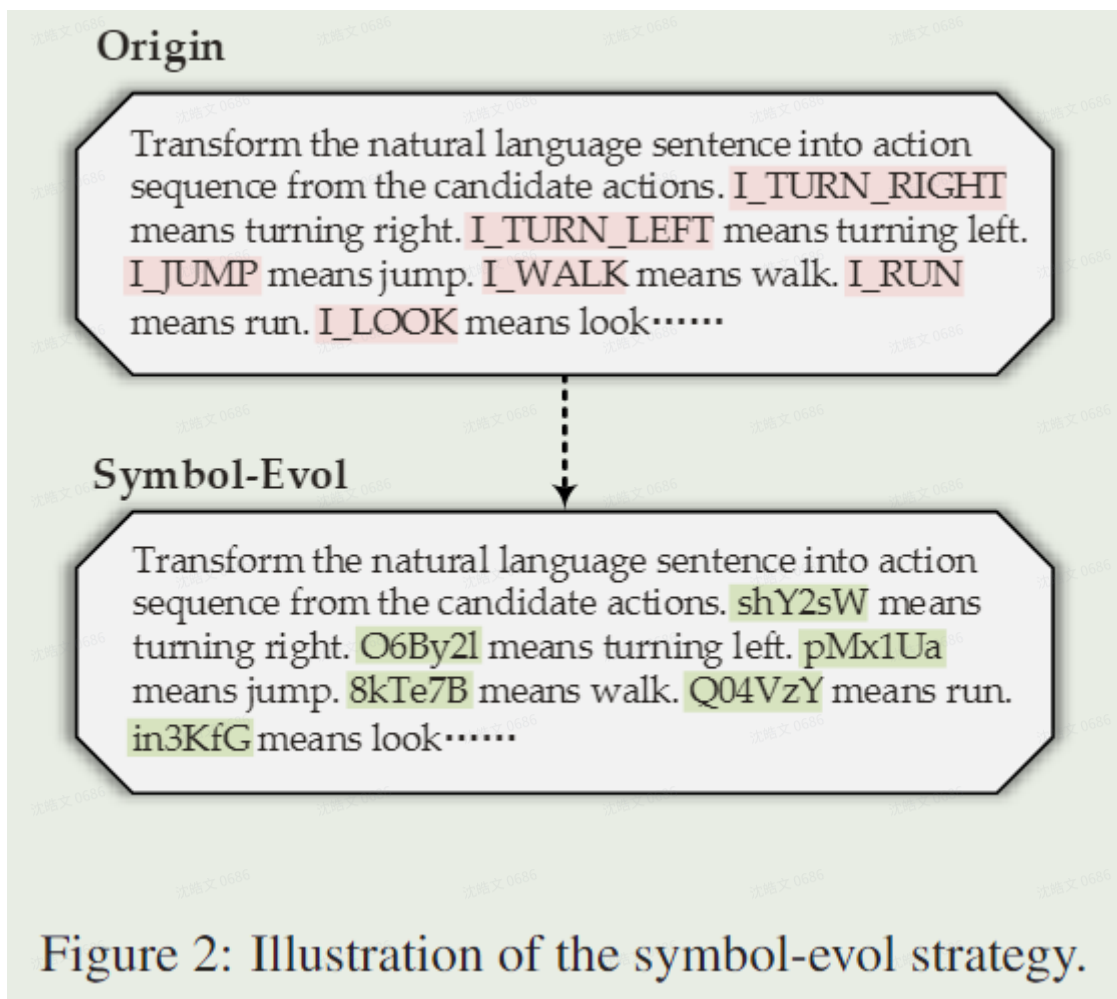
1. Motivation

1. Insufficient performance of large language models on **symbol center tasks** .
2. Existing solutions
 - a. Fine-tuning Pre-trained Models: Catastrophic Forgetting for Generic Capabilities
 - b. Existing methods of injecting symbols
 - i. Complex, time-consuming, and labor-intensive
 - ii. Process each symbol independently, ignoring the interaction between symbols

2. Contributions

Two approaches: data and frameworks, **boosting symbol-centric tasks** , **balancing symbolic and natural language abilities (general ability)**

1. Data aspects
 - a. Symbolic Data Collection
 - i. Collect data for 34 **text-to-symbol generation tasks** using 3 symbolic data collection methods
 1. Generate from existing benchmark: convert original data source to instruction format
 2. Generate text-to-symbol pairs by giving LLM prompts
 3. Introducing **Symbol-evol strategy** to reduce the **memory pattern** tendency of large models



ii. Data set composition: $D'_s = D_s \cup D_{evol}$

1. D_s : Collection of data collected by methods 1 and 2
2. D_{evol} : For the D_s dataset obtained after using the Symbol-evol strategy
3. D'_s : The **intersection**
4. D''_s **Subset** D'_s of:

b. General Data Collection

i. Through **3** general **instruction data** collection methods

1. Sampled **Flan** collection data
2. **Code Alpaca** instruction tuning data
3. Sampled Evol - data from **WizardLM**

ii. Data set composition D_g

2. Framework: Two-stage Tuning Framework

Overall Route: LLAMA2-Chat - (Injection stage) -> Symbol-LLMBase - (Infusion stage) -> Symbol-LLMInstruct

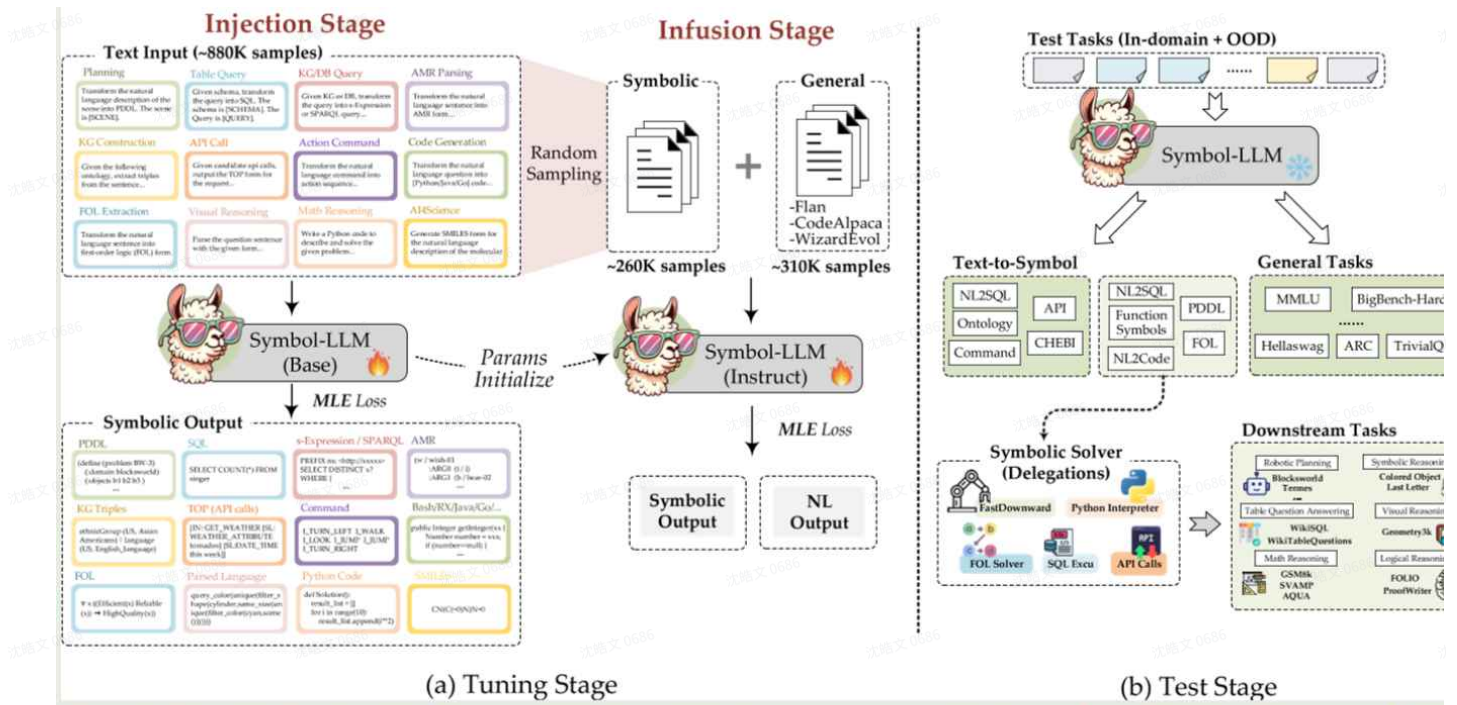


Figure 1: Model structure of Symbol-LLM. (a) is two training stages, Injection stage and Infusion stage. (b) is the test stage with three parts of tasks, text-to-symbol generation tasks, general tasks and downstream tasks under the Symbol+Delegation setting 3.

a. Injection stage : effectively injecting symbolic knowledge

- Use D'_s for training ~ 880K samples
- Goodness function: MLE

b. Infusion stage : mitigates catastrophic forgetting of general abilities

- Use D'_s and D_g train to uniformly sample at a ratio of 0.3 to obtain a sample set of ~ 260K
- Goodness function: MLE

3. Experiments

The superiority of the Symbol-LLM series has been proven

1. 34 Text-to-Symbol Generation Task Results

Domains / Tasks		Metrics	Close-Source		Symbol LLM			
			ChatGPT	Claude	7B-Base	7B-Instruct	13B-Base	13B-Instruct
Planning	Blocksworld	BLEU	96.54	91.35	98.41	99.02	98.49	99.02
	Termes	BLEU	74.73	26.94	90.11	48.69	89.85	90.09
	Floortile	BLEU	54.23	13.94	96.19	95.84	95.58	95.24
	Grippers	BLEU	99.90	90.91	99.12	98.53	99.00	98.89
SQL	Spider	EM	42.60	32.70	66.90	63.80	69.10	69.20
	Sparc	EM	29.90	28.60	56.50	55.00	58.80	58.90
	Cosql	EM	18.80	22.70	51.50	48.20	53.40	52.70
KG / DB	WebQSP	F1	36.49	41.37	83.90	84.43	86.07	85.29
	GrailQA	EM	28.52	25.56	80.09	79.24	80.67	81.17
	CompWebQ	EM	0.00	0.00	51.20	50.98	55.62	54.94
AMR	AMR3.0	Smatch	18.00	10.00	55.00	54.00	54.00	55.00
	AMR2.0	Smatch	14.00	12.00	47.00	45.00	46.00	46.00
	BioAMR	Smatch	23.00	3.00	78.00	78.00	79.00	80.00
Ontology	Tekgen	F1	8.92	1.86	58.76	57.34	60.27	58.55
	Webnlg	F1	28.34	8.89	63.67	60.42	65.28	63.08
API	MTOP	EM	3.80	8.40	84.00	84.40	86.20	86.60
	TOPv2	EM	6.60	7.60	87.00	85.80	86.60	85.20
	NLmaps	EM	30.88	16.77	92.60	92.18	92.45	92.21
Command	SCAN	EM	15.09	15.97	98.49	98.35	99.43	99.28
Code	NL2BASH	BLEU	54.19	42.24	58.46	60.25	61.00	60.76
	NL2RX	BLEU	38.60	18.30	85.32	85.08	85.71	84.97
	NL2Python	BLEU	37.01	36.73	37.23	39.79	39.61	40.76
	NL2Java	BLEU	24.88	22.79	26.93	28.08	27.65	28.25
	NL2Go	BLEU	19.08	26.65	22.90	29.19	28.66	30.31
FOL	FOLIO	LE	60.65	53.47	90.25	90.58	90.19	90.65
	MALLS	LE	69.15	30.46	88.43	88.88	89.11	89.50
	LogicNLI	LE	73.11	69.16	99.98	99.97	99.98	100.00
Visual	GQA	EM	7.55	7.70	85.50	85.50	85.25	85.95
	CLEVR	EM	6.35	5.90	95.50	94.80	74.25	95.60
	Geometry3k	EM	65.25	40.84	94.52	95.13	95.21	95.67
Math	GSM8K-Code	BLEU	88.65	71.60	88.39	74.95	88.87	92.42
	AQUA-Code	BLEU	68.25	49.48	67.44	67.21	68.28	68.44
	MATH-Code	BLEU	67.48	66.25	69.33	68.38	73.45	64.06
AI4Science	CheBi-20	EM	1.15	0.30	53.30	58.97	60.70	65.27
Average Performance			28.22	21.85	73.44	72.11	73.97	74.81

Table 1: Main results on 34 text-to-symbol generation tasks. The better results with the same model size are marked in bold.

- Better results for the same model size are marked in bold

2. General capabilities

Models	MMLU (5-shot)					BBH (0-shot)
	Humanities	SocialSciences	STEM	Others	Average	Average
Open-source LLMs (7B)						
LLaMA-2	<u>42.47</u>	<u>52.49</u>	<u>36.94</u>	<u>52.47</u>	<u>45.78</u>	<u>35.01</u>
Symbol-LLM _{Base}	40.04	46.28	33.73	47.16	41.70	33.82
Symbol-LLM _{Instruct}	46.33	57.20	40.39	54.53	49.30	39.30
Open-source LLMs (13B)						
LLaMA-2	49.52	62.43	43.84	60.02	53.55	<u>36.99</u>
Symbol-LLM _{Base}	45.67	55.67	40.09	53.89	48.56	35.26
Symbol-LLM _{Instruct}	<u>48.88</u>	<u>62.14</u>	<u>43.44</u>	<u>57.93</u>	<u>52.71</u>	44.09

Table 2: Results on General Tasks. We include 57 tasks in MMLU benchmark for testing under the 5-shot setting (Hendrycks et al., 2021a), while we select 21 tasks in Bigbench-Hard benchmark under the 0-shot setting following (Gao et al., 2021).

3. Math Reasoning

Models	Del.	GSM8k	MATH	GSM-Hard	SVAMP	ASDiv	ADDSUB	SingleEQ	SingleOP	MultiArith
Is OOD Setting		✗	✗	✓	✓	✓	✓	✓	✓	✓
Close-source LLMs										
GPT-3.5	✓	4.60	1.05	4.62	5.10	6.30	1.01	3.94	8.54	17.33
GPT-3.5 (3-shot)	✓	76.04	36.80	62.09	83.40	85.73	87.59	96.46	90.74	96.67
Claude-1	✓	11.14	1.07	9.02	10.30	6.30	5.06	4.53	0.36	12.67
Claude-1 (3-shot)	✓	58.07	13.17	43.75	78.90	74.19	79.49	88.19	87.72	91.83
Open-source LLMs (7B)										
LLaMA-2-Chat (3-shot)	✓	12.21	1.32	10.69	22.00	25.86	29.11	27.36	39.15	23.17
WizardMath†	✗	54.90	10.70	-	57.30	-	-	-	-	-
MAmmoTH†	✓	51.70	31.20	-	66.70	-	-	-	-	-
Symbol-LLM _{Base}	✓	61.14	<u>28.24</u>	52.62	<u>72.50</u>	78.34	89.62	97.83	96.26	99.67
Symbol-LLM _{Instruct}	✓	<u>59.36</u>	26.54	<u>48.98</u>	72.80	<u>75.76</u>	<u>87.85</u>	<u>96.26</u>	<u>93.24</u>	<u>99.00</u>
Open-source LLMs (13B)										
LLaMA-2-Chat (3-shot)	✓	34.87	6.07	28.96	45.00	46.61	45.57	47.05	56.76	56.67
WizardMath†	✗	63.90	14.00	-	64.30	-	-	-	-	-
MAmmoTH†	✓	61.70	36.00	-	72.40	-	-	-	-	-
Symbol-LLM _{Base}	✓	68.69	33.39	58.53	78.80	80.15	<u>91.14</u>	96.85	95.55	98.83
Symbol-LLM _{Instruct}	✓	<u>65.58</u>	31.32	<u>55.57</u>	<u>76.80</u>	<u>79.01</u>	91.90	96.85	<u>94.84</u>	99.33

Table 3: Results on Math Reasoning. Del. represents whether uses delegation. Results are under the zero-shot setting unless otherwise stated (following Table 4, 5, 6 and 7 share the same setting). † means results are directly derived from the original paper (Luo et al., 2023; Yue et al., 2023)

4. Symbolic Reasoning

Models	Del.	ColoredObject	LastLetter
Is OOD Setting		✓	✓
Close-source LLMs			
GPT-3.5	✓	12.45	94.00
Claude-1	✓	46.05	90.67
Open-source LLMs (7B)			
LLaMA-2-Chat	✓	28.70	0.00
Symbol-LLM _{Base}	✓	22.65	90.67
Symbol-LLM _{Instruct}	✓	<u>25.50</u>	96.67
Open-source LLMs (13B)			
LLaMA-2-Chat	✓	30.35	0.00
Symbol-LLM _{Base}	✓	36.35	94.00
Symbol-LLM _{Instruct}	✓	<u>34.00</u>	96.67

Table 4: Results on Symbolic Reasoning.

5. Logical Reasoning

Models	Del.	FOLIO	ProofWriter	PrOntoQA
Is OOD Setting		✗	✗	✓
Close-source LLMs				
GPT-3.5	✓	44.61	29.00	52.00
Claude-1	✓	37.25	35.83	55.80
Logic-LM (SOTA)	✓	61.76	70.11	93.20
Open-source LLMs (7B)				
LLaMA-2-Chat	✓	34.80	34.83	50.00
Symbol-LLM _{Base}	✓	<u>46.08</u>	76.50	<u>55.60</u>
Symbol-LLM _{Instruct}	✓	49.02	<u>76.33</u>	57.20
Open-source LLMs (13B)				
LLaMA-2-Chat	✓	33.33	35.83	49.20
Symbol-LLM _{Base}	✓	<u>33.82</u>	76.33	48.40
Symbol-LLM _{Instruct}	✓	35.29	<u>75.50</u>	53.60

Table 5: Results on Logical Reasoning. All results are obtained under the one-shot setting.

6. Comparison between single SFT and unified SFT

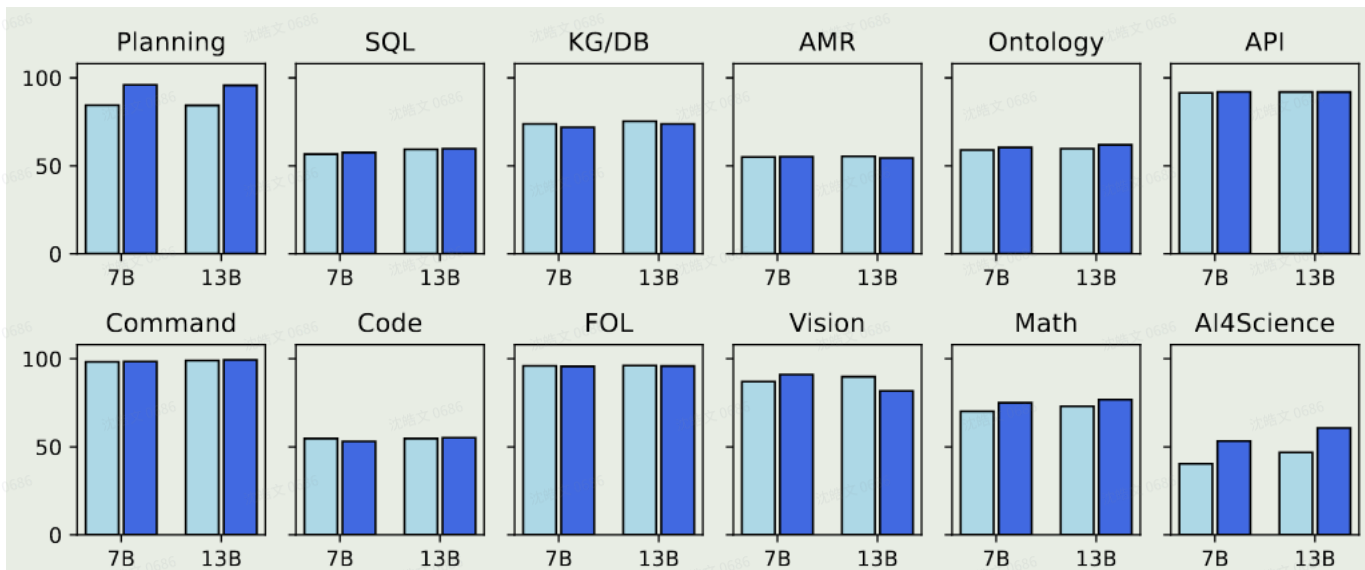


Figure 5: Comparison between single SFT and unified SFT.

- Single SFT (Single Supervised Fine-tuning): Fine-tuning a large pre-trained model for a specific task or dataset
- Unified SFT (Unified Supervised Fine-tuning): Tuning the model uniformly so that it performs well on multiple tasks, rather than just optimizing for a single task
- Light blue bars represent single-domain SFTs, and dark blue bars represent unified SFTs. In most fields, unified SFTs are better than single-domain SFTs, indicating that **there are potential relationships between various symbols**

MAMMOTH

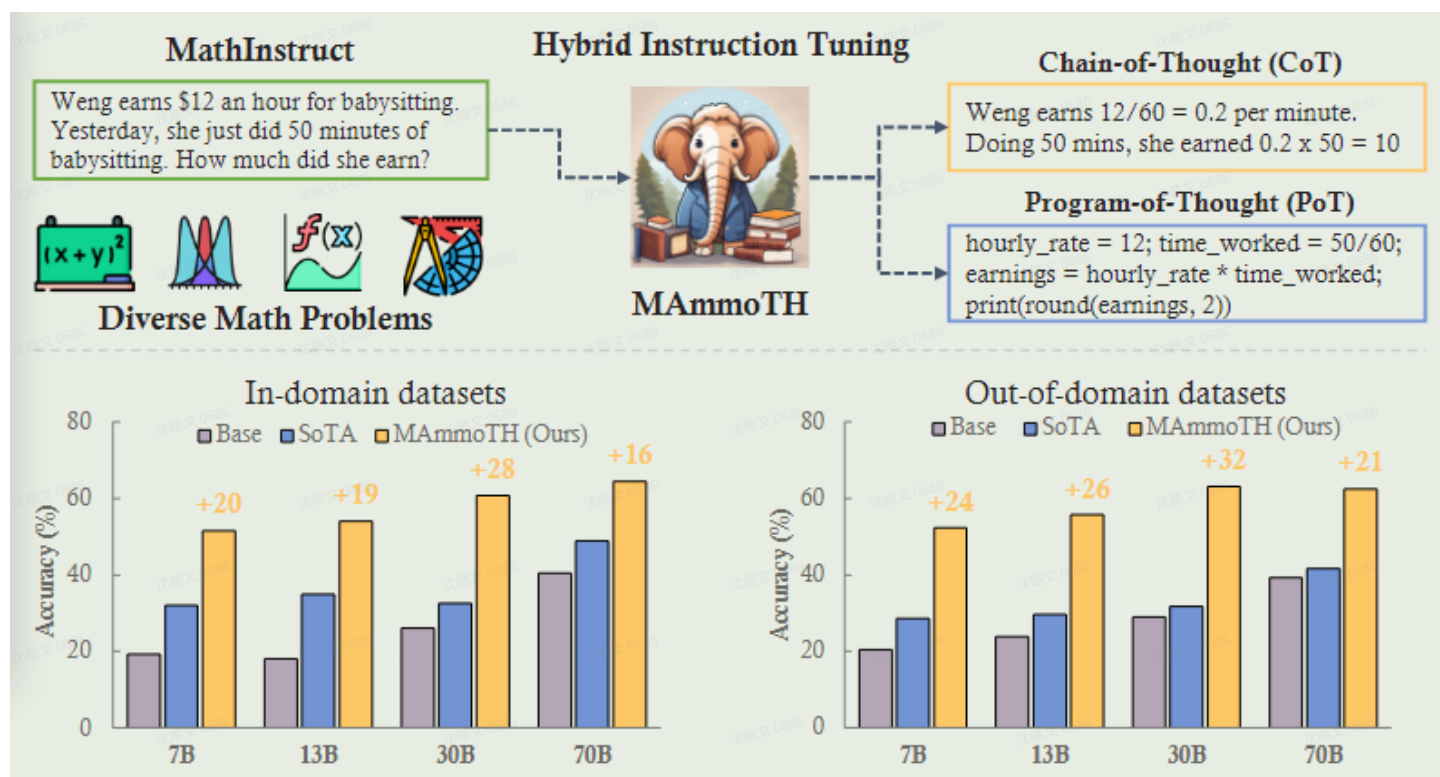
1. Motivation

1. There is a significant gap in **mathematical reasoning ability** between open source large models and closed source large models

2. Current solutions

- a. More **pre-training** : higher computational costs
- b. **Fine-tuning** on specific datasets (mathematical instructions): poor generalization
 - Existing **Mathematical Instruction Adjustment Methods** and Their **Disadvantages**
 - CoT (Chain-of-Thought): Poor in **computational accuracy, complex mathematical problems, and algorithmic reasoning**.
 - PoT (Program-of-Thought): Poor performance in **abstract reasoning scenarios**

2. Contributions



1. Created a **command tuning dataset** : **MathInstruct**

a. Coverage **of different mathematical domains** and **levels of complexity**

- Narrow the selection of data to some widely used high-quality datasets: GSM8K, MATH, AQUA, Camel, and TheoremQA

b. The **integration of CoT and PoT**

- CoT: Use **GPT-4** to synthesize **CoT rationales** for problems in TheoremQA, create question-CoT pairs through Self-Instruct
- PoT: Use **GPT-4** to supplement the **PoT rationales** of the selected dataset, compare the performance of these GPT-4 synthesizers with the basic facts of human annotation to filter these GPT-4 synthesizers

2. Using MathInstruct Instruction Set Architecture to **fine-tune Llama** base models at different scales from 7B to 70B

3. Evaluation method

a. By default, the model provides a CoT solution

In order to switch to the PoT method, you can add the trigger phrase after the question **"Let's write a program to solve this problem"**

- Afterwards, a simple **hybrid decoding strategy** is introduced: the model first tries PoT hints. If the program is not executable, we fall back to CoT hints

3. Experiments

1. Dataset selection

Eval Dataset	# Samples	In-Domain?	Answer Form	Fields
GSM8K (Cobbe et al., 2021)	1319	YES	Open-formed	■
MATH (Hendrycks et al., 2021b)	5000	YES	Open-formed	■ ■ ■ ■ ■ ■ ■ ■
AQuA-RAT (Ling et al., 2017)	254	YES	Multi-choice	■
NumGLUE (Mishra et al., 2022b)	1042	YES	Open-formed	■
SVAMP (Patel et al., 2021)	1000	NO	Open-formed	■
Mathematics (Davies et al., 2021)	1000	NO	Open-formed	■ ■ ■ ■ ■
SimulEq (Koncel-Kedziorski et al., 2016)	514	NO	Open-formed	■
SAT-Math (Zhong et al., 2023)	220	NO	Multi-choice	■ ■ ■
MMLU-Math (Hendrycks et al., 2021a)	974	NO	Multi-choice	■ ■ ■ ■ ■

Table 2: Comprehensive overview of our evaluation datasets, featuring a variety of in-domain and out-of-domain problems across diverse fields of mathematics. Different colored squares represent different fields in mathematics: ■ Pre-Algebra; ■ Inter-Algebra; ■ Algebra; ■ Probability; ■ NumTheory; ■ Calculus; ■ Geometry.

2. Domain evaluation results

Model	Base	Math-SFT?	GSM8K	MATH	AQuA	NumGLUE	Avg
Closed-source Model							
GPT-4	-	Unknown	92.0 [†]	42.5 [†]	72.6 [†]	-	-
GPT-4 (Code-Interpreter)	-	Unknown	97.0 [†]	69.7 [†]	-	-	-
PaLM-2	-	Unknown	80.7 [†]	34.3 [†]	64.1	-	-
Claude-2	-	Unknown	85.2 [†]	32.5 [†]	60.9	-	-
Codex (PoT)	-	No	71.6 [†]	36.8 [†]	54.1 [†]	-	-
ART (InstructGPT)	-	Unknown	71.0	-	54.2	-	-
7B Parameter Model							
Llama-1	-	No	10.7 [†]	2.9 [†]	22.6	24.7	15.5
Llama-2	-	No	14.6 [†]	2.5 [†]	30.3	29.9	19.3
Galactica-6.7B	GAL	GAL-Instruct	10.2	2.2	25.6	25.8	15.9
Code-Llama (PoT)	-	No	25.2	13.0	24.0	26.8	22.2
AQuA-SFT	Llama-2	AQuA	11.2	3.6	35.6	12.2	15.6
Llama-1 RFT	Llama-1	GSM8K	46.5 [†]	5.2	18.8	21.1	22.9
WizardMath	Llama-2	GSM8K+MATH	54.9 [†]	10.7 [†]	26.3	36.1	32.0
MAmmoTH	Llama-2	MathInstruct	53.6	31.5	44.5	61.2	47.7
MAmmoTH-Coder	Code-Llama	MathInstruct	59.4	33.4	47.2	66.4	51.6
Δ			+5	+21	+12	+30	+20
13-15B Parameter Model							
Llama-1	-	No	17.8 [†]	3.9 [†]	26.0	24.8	18.1
Llama-2	-	No	28.7 [†]	3.9 [†]	25.1	8.8	16.6
Code-Llama (PoT)	-	No	36.1	16.4	28.7	29.2	27.6
CodeT5+ (PoT)	-	No	12.5	2.4	20.5	19.4	13.7
CodeGen+ (PoT)	-	No	12.7	3.4	24.5	22.5	15.7
Vicuna-1.5	Llama-2	No	28.4 [†]	5.8	24.8	36.9	23.9
Llama-1 RFT	Llama-1	GSM8K	52.1 [†]	5.1	16.1	24.5	24.4
Orca-Platypus	Llama-2	Platypus	38.4	3.0	18.9	35.3	23.9
Platypus	Llama-2	Platypus	25.7	2.5	33.4	42.3	25.9
WizardMath	Llama-2	GSM8K+MATH	63.9 [†]	14.0 [†]	21.2	40.8	34.9
MAmmoTH	Llama-2	MathInstruct	62.0	34.2	51.6	68.7	54.1
MAmmoTH-Coder	Code-Llama	MathInstruct	64.7	36.3	46.9	66.8	53.7
Δ			+1	+20	+18	+26	+19
30-34B Parameter Model							
Llama-1	-	No	35.6 [†]	7.1 [†]	33.4	28.4	26.1
Code-Llama (PoT)	-	No	44.0	23.1	25.2	29.3	30.4
Llama-1 RFT	Llama-1	GSM8K	56.5 [†]	7.4 [†]	18.5	24.3	26.6
Galactica-30B	GAL	GAL-Instruct	41.7	12.7	28.7	34.7	29.4
Platypus	Llama-1	Platypus	37.8	9.3	27.9	40.5	28.8
Tulu	Llama-2	Tulu	51.0	10.8	25.5	43.4	32.6
MAmmoTH-Coder	Code-Llama	MathInstruct	72.7	43.6	54.7	71.6	60.7
Δ			+16	+21	+21	+28	+28
65-70B Parameter Model							
Llama-1	-	No	50.9 [†]	10.6 [†]	35.0	50.2	36.6
Llama-2	-	No	56.8 [†]	13.5 [†]	40.9	50.4	40.4
Llama-2-Chat	Llama-2	No	54.9	18.6	37.0	51.6	40.5
Guamaco	Llama-2	No	59.2	4.1	45.2	53.5	40.5
WizardMath	Llama-2	GSM8K+MATH	81.6 [†]	22.7 [†]	20.0	48.9	43.3
Platypus	Llama-2	Platypus	70.6	15.6	51.2	55.4	48.1
MAmmoTH	Llama-2	MathInstruct	76.9	41.8	65.0	74.4	64.5
Δ			-5	+19	+14	+19	+16

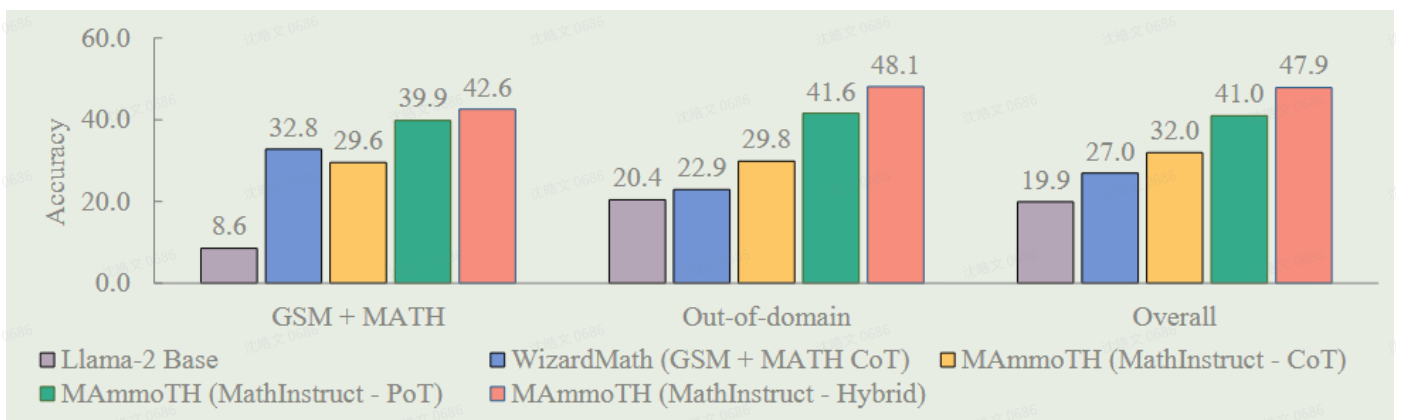
- Performance on 9 mathematical inference datasets is significantly better than existing open source models, and even outperforms closed source models in some tasks
- MAMmoTH can achieve universal improvements comprehensively
- MAMmoTH is particularly strong in solving **more complex mathematical problems** in MATH

3. Extraterritorial evaluation results

Model	SVAMP	Mathematics	SimulEq	SAT-Math	MMLU-Math	Avg
Closed-source Model						
GPT-4	97.0 [†]	-	-	95 [†]	-	-
Codex (PoT)	85.2 [†]	-	-	68 [†]	-	-
7B Parameter Model						
Llama-1	24.5	6.2	4.6	22.7	30.6	17.7
Llama-2	34.5	6.0	5.0	26.8	29.8	20.4
Code-Llama (PoT)	49.4	21.7	3.5	28.6	26.9	26.0
Llama-1 RFT	21.1	5.1	11.0	12.5	21.7	14.3
Galactica-6.7B	25.6	4.6	4.2	17.5	28.0	16.0
WizardMath	36.1	9.3	12.8	25.4	31.1	28.6
Toolformer	29.4 [†]	-	-	-	-	-
MAMmoTH	67.7	46.3	41.2	42.7	42.6	48.1
MAMmoTH-Coder	71.4	55.4	45.9	40.5	48.3	52.3
Δ	+22	+34	+33	+14	+17	+24
13B Parameter Model						
Llama-1	34.7	6.9	5.4	27.7	30.7	21.0
Llama-2	35.1	11.5	5.8	32.7	34.4	23.9
Code-Llama (PoT)	60.0	21.3	3.8	25.9	27.7	27.7
Vicuna-1.5	55.7	10	6.6	34.0	34.1	28.1
Llama-1 RFT	46.5	6.7	10.1	13.2	21.6	19.6
WizardMath	51.9	14.1	14.9	24.5	32.1	27.5
Platypus	55.4	11.4	7.4	36.8	35.5	29.3
Orca-Platypus	56.8	12.6	7.9	29.5	41.6	29.7
MAMmoTH	72.4	49.2	43.2	46.8	47.6	51.8
MAMmoTH-Coder	73.7	61.5	47.1	48.6	48.3	55.8
Δ	+14	+40	+33	+12	+7	+26
30-34B Parameter Model						
Llama-1	48.8	12.8	11.2	33.4	39.0	29.0
Code-Llama (PoT)	69.1	34.5	6.8	26.8	21.6	31.7
Llama-1 RFT	55.4	7.6	12.8	20.4	37.9	26.8
Galactica-30B	41.6	11.8	13.2	37.7	37.9	28.4
Tulu	59.0	10.7	10.3	31.3	39.8	30.2
Platypus	51.7	13.8	13.6	38.6	41.0	31.7
MAMmoTH-Coder	84.3	65.4	51.8	60.9	53.8	63.2
Δ	+15	+31	+38	+22	+13	+32
65-70B Parameter Model						
Llama-1	55.3	14.2	15.2	37.4	44.1	33.2
Llama-2	63.8	20.5	14.0	51.3	47.1	39.3
Llama-2-Chat	71.5	19.2	21.7	44.1	46.9	40.6
WizardMath	71.8	17.1	37.9	13.2	27.4	33.4
Guanaco	66.8	17.8	20.2	50.0	47.3	40.4
Platypus	51.8	26.3	21.7	55.9	52.5	41.6
MAMmoTH	82.4	55.6	51.4	66.4	56.7	62.5
Δ	+11	+29	+14	+11	+4	+21

- Strong generalization ability: the performance improvement on the OOD dataset is **more significant**, proving the **significant universality**

4. Ablation of the Data Source



- CoT subsets help maintain general language-based reasoning skills to handle scenarios that PoT cannot handle well
- PoT subsets can teach models how to solve complex math problems with high precision using the Python API

5. Influence of Major Subsets

Gradually add each dataset to the training and compare the performance with the dataset fine-tuned on the entire MathInstruct

Training Data	GSM	MATH	AQuA	NumG	SVA	Mat	Sim	SAT	MMLU	AVG
-	14.6	2.5	30.3	29.9	34.5	6.0	5.0	26.8	29.8	-25.3
G	56.6	9.2	24.4	32.1	65.4	20.5	12.3	27.2	25.2	-22.7
G + M	58.1	28.2	26.0	34.7	64.8	50.1	17.1	28.6	28.4	-19.5
G + M + C	57.4	28.5	26.2	37.5	65.3	50.4	17.7	29.3	28.7	-19.2
G + M + C + A	57.5	29.1	46.9	42.2	65.8	49.6	32.7	42.3	43.1	-4.8
G + M + C + A + N	56.5	28.9	38.2	63.7	64.1	47.9	40.8	38.6	44.5	-3.4
Existing Data	31.4	18.4	40.3	53.3	61.8	27.9	45.6	32.7	38.4	-9.0
MathInstruct	53.6	31.5	44.5	61.2	67.7	46.3	41.2	42.7	42.6	47.9

- When the data **is not very diverse** at the beginning of training, the overall generalization performance is very poor
- As other major subsets are gradually added, MAMmoTH can be observed to become a better **mathematical generalist**
- The newly planned data significantly improves the performance of the model on many datasets

WizardMath

1. Motivation

- Most existing open-source models are only pre-trained on large-scale internet data, without optimization for mathematical relevance, and these models are difficult to solve **complex multi-step quantitative inference**

2. Contributions

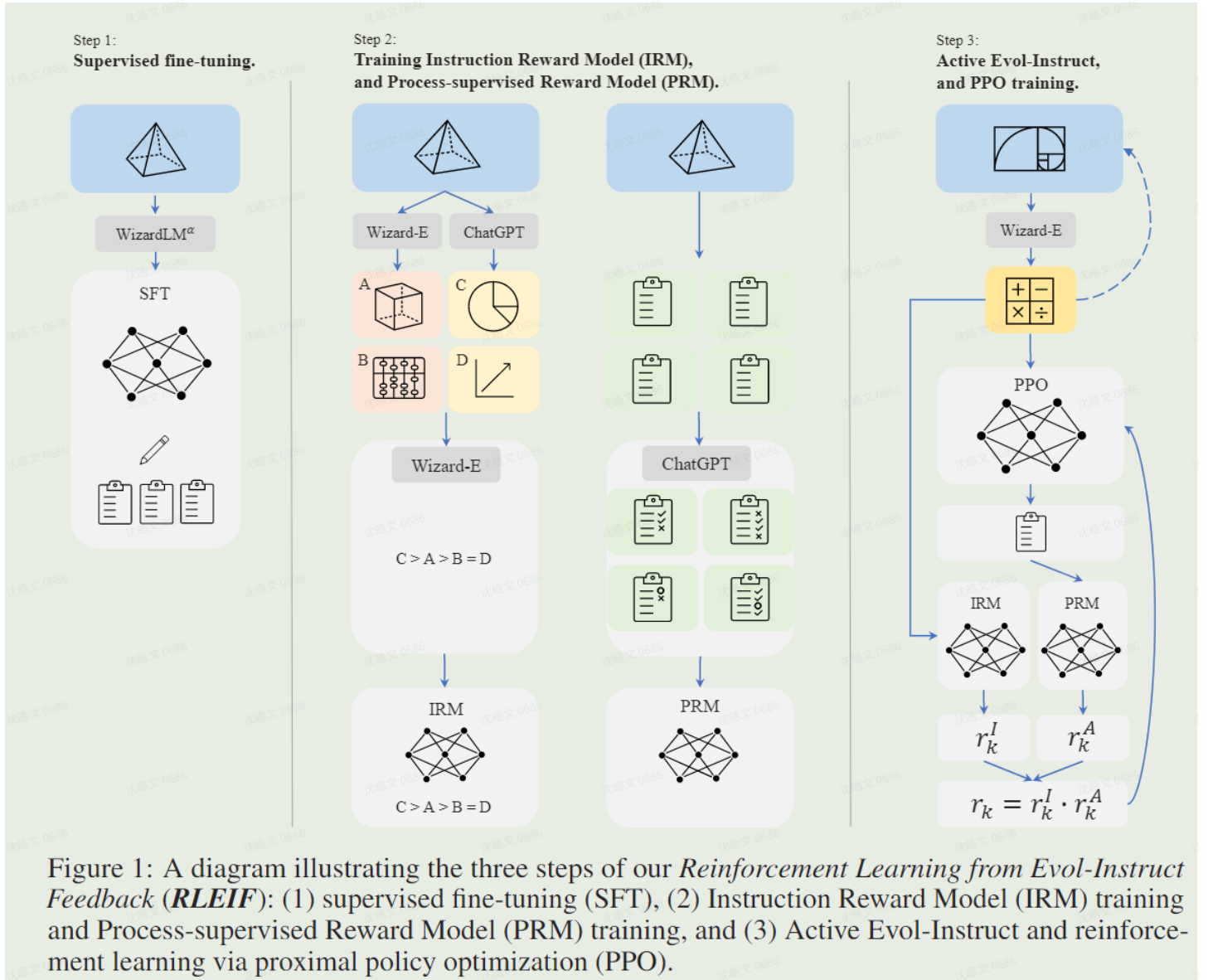


Figure 1: A diagram illustrating the three steps of our *Reinforcement Learning from Evol-Instruct Feedback (RLEIF)*: (1) supervised fine-tuning (SFT), (2) Instruction Reward Model (IRM) training and Process-supervised Reward Model (PRM) training, and (3) Active Evol-Instruct and reinforcement learning via proximal policy optimization (PPO).

1. Supervised fine-tuning

- Use the **Alpha version of the WizardLM 70Bll model to generate 15k answers for GSM8k and MATH**, generate solutions in step-by-step format, use this data to fine-tune the base Llama model
 - We extracted **1.5k open domain dialogues** from the training data of WizardLM and then merged them with the above mathematical corpus as the final SFT training data
- ### 2. Through the adjusted Evol-Instruct method to generate a variety of mathematical instruction data

a. Evol-Instruct method



Figure 1: Running Examples of *Evol-Instruct*.

i. In-depth Evolving (blue direction line)

- Add constraints
- Deepen
- Concretization
- Add inference step
- Complicate the input

ii. In-breadth Evolving (red direction line)

- Mutation: Generate brand new instructions based on given instructions

iii. Elimination Evolving: Use an instruction eliminator to **filter out failed** instructions

b. **Downward evolution** : Generating **simpler** questions

- Modify high-difficulty questions to low-difficulty questions

- Create a new and easier question with a different theme.
- **Upward evolution** : Generating **harder** problems
 - Add more constraints
 - Concretization
 - Increase reasoning
- a. **Result** : Number of questions (after 8 turns): **15k -- > 96k**
- 3. Train an **instruction reward model IRM** and a **process supervised reward model PRM**
 - a. **IRM** : Judging the quality of evolutionary instructions

Specific method: Use ChatGPT and Wizard-E to generate 2-4 instructions, and then use the Wizard-E model to evaluate

 - Evaluate instructions from three perspectives
 - Definition
 - Intensive reading
 - Integrity
 - b. **PRM** : Supervisory Process (Judgment Process)

Specific method: rely on ChatGPT to provide process supervision and require it to evaluate the correctness of each step in the solution generated by our model
- 4. Using generated **mathematical instruction data** for **PPO reinforcement learning** through **IRM and PRM models**

3. Experiments

1. Comparison with base line models (open source, closed source)

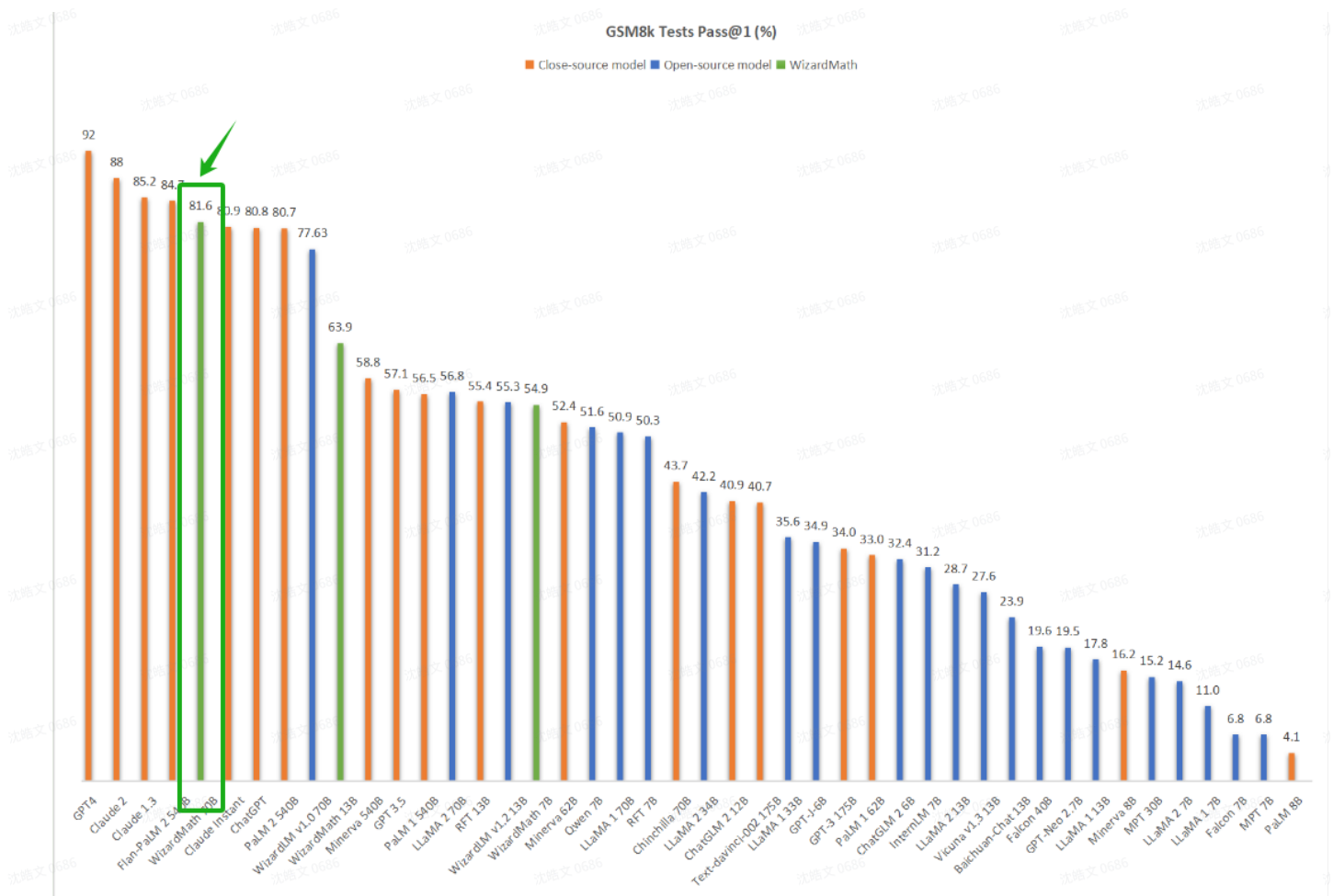
Greedy decoding and CoT

Model	Params	GSM8k	MATH
Closed-source models			
GPT-4 [3]	-	92.0	42.5
Claude 2 [7]	-	88.0	-
Claude 1.3 [7]	-	85.2	-
Flan-PaLM 2 [44]	540B	84.7	33.2
Claude Instant [7]	-	80.9	-
ChatGPT [46]	-	80.8	34.1
PaLM 2 [44]	540B	80.7	34.3
	8B	16.2	14.1

Minerva [15]	62B	52.4	27.6
	540B	58.8	33.6
GPT-3.5 [3]	-	57.1	-
PaLM [7]	8B	4.1	1.5
	62B	33.0	4.4
	540B	56.5	8.8
RFT-13B [16]	13B	55.4	-
Chinchilla [47]	70B	43.7	-
ChatGLM 2 [45]	12B	40.9	-
Text-davinci-002 [15]	175B	40.7	19.1
GPT-3 [1]	175B	34.0	5.2
GPT-2 [43]	1.5B	-	6.9
Open-source models			
GAL [14]	30B	-	12.7
	120B	-	20.4
LLaMA 2 [20]	7B	14.6	2.5
	13B	28.7	3.9
	34B	42.2	6.24
	70B	56.8	13.5
Qwen [10]	7B	51.6	-
LLaMA 1 [4]	7B	11.0	2.9
	13B	17.8	3.9
	33B	35.6	7.1
	65B	50.9	10.6
RFT-7B [16]	7B	50.3	-
GPT-J-6B [48]	6B	34.9	-
ChatGLM 2 [45]	6B	32.4	-
InternLM-7B [49]	7B	31.2	-
Vicuna v1.3 [23]	13B	27.6	-
Baichuan-chat [9]	13B	23.9	-
Falcon [21]	7B	6.8	2.3
	40B	19.6	2.5
GPT-Neo-2.7B [50]	2.7B	19.5	-
MPT [8]	7B	6.8	3.0
	30B	15.7	3.1

	JOB	13.2	5.1
WizardMath	7B	54.9 (+3.3)	10.7 (+7.7)
WizardMath	13B	63.9 (+35.2)	14.0 (+10.1)
WizardMath	70B	81.6 (+24.8)	22.7 (+9.2)

- WizardMath 7B surpasses most open source models with a parameter count of about 7B to 40B.
- WizardMath 13B is significantly better than Llama 1 65B and Llama 2 70B on GSM8k. In addition, it is significantly better than Llama 1 65B and Llama 2 70B on MATH.
- WizardMath 70B greatly surpasses Llama 2 70B on GSM8k. At the same time, its performance on MATH is also better than Llama 2 70B



- Currently ranked in the top five among all models

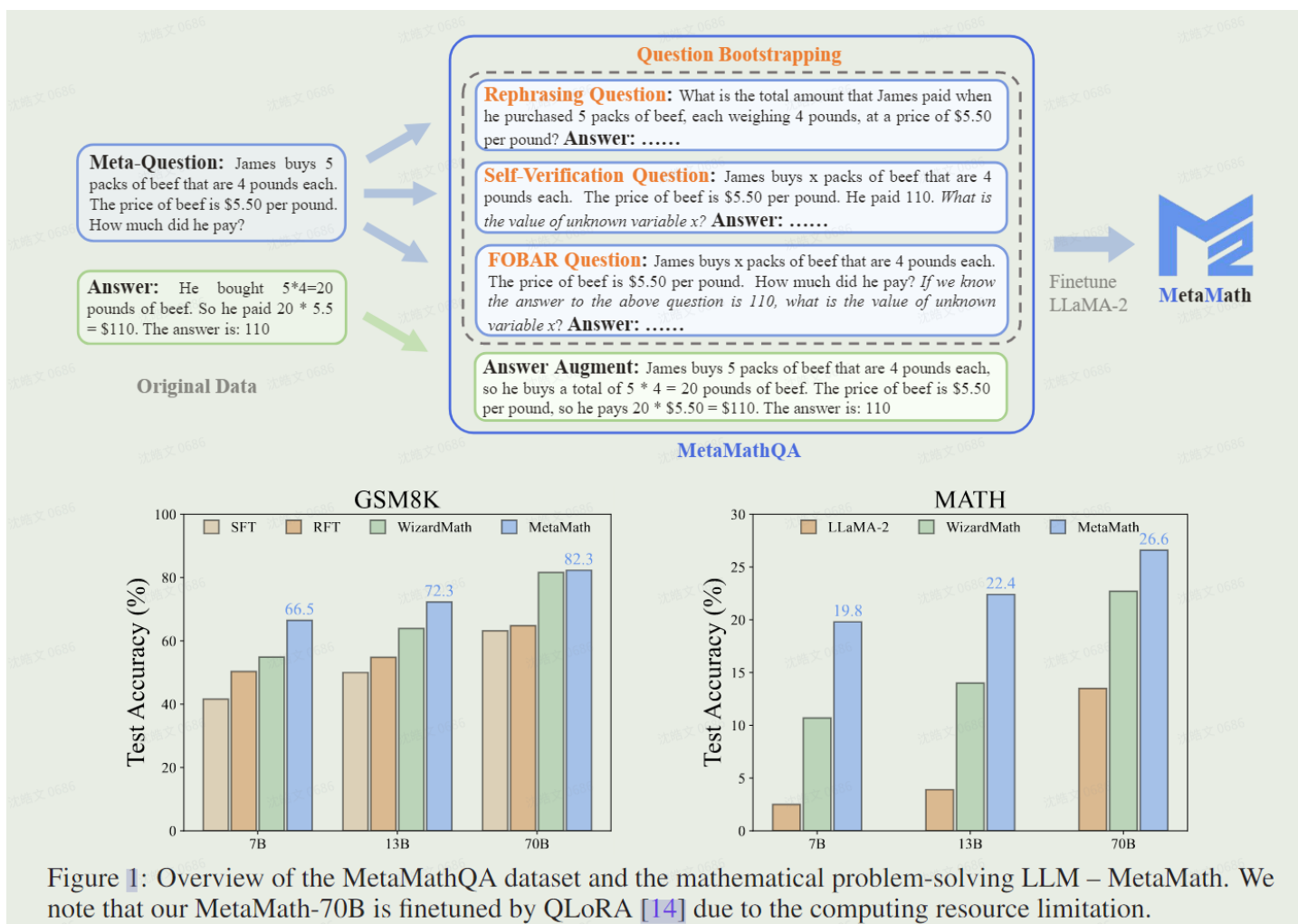
MetaMath

1. Motivation

Large language models have the phenomenon of "**reverse curse**", that is, a language model trained on "A is B" cannot be generalized to "B is A"

- Reason: The commonly used mathematical reasoning dataset has **limited sample size** and **insufficient problem diversity**

2. Contributions



1. Based on two commonly used mathematical datasets (GSM8K and MATH), the **MetaMathQA dataset**

- Composed of 395K **large language models generating** forward and reverse mathematical question pairs
- Constructed a reverse inference dataset GSM8K-Backward
- This dataset is rich in diversity: through **question guidance**, the MetaMathQA dataset is more **diverse**

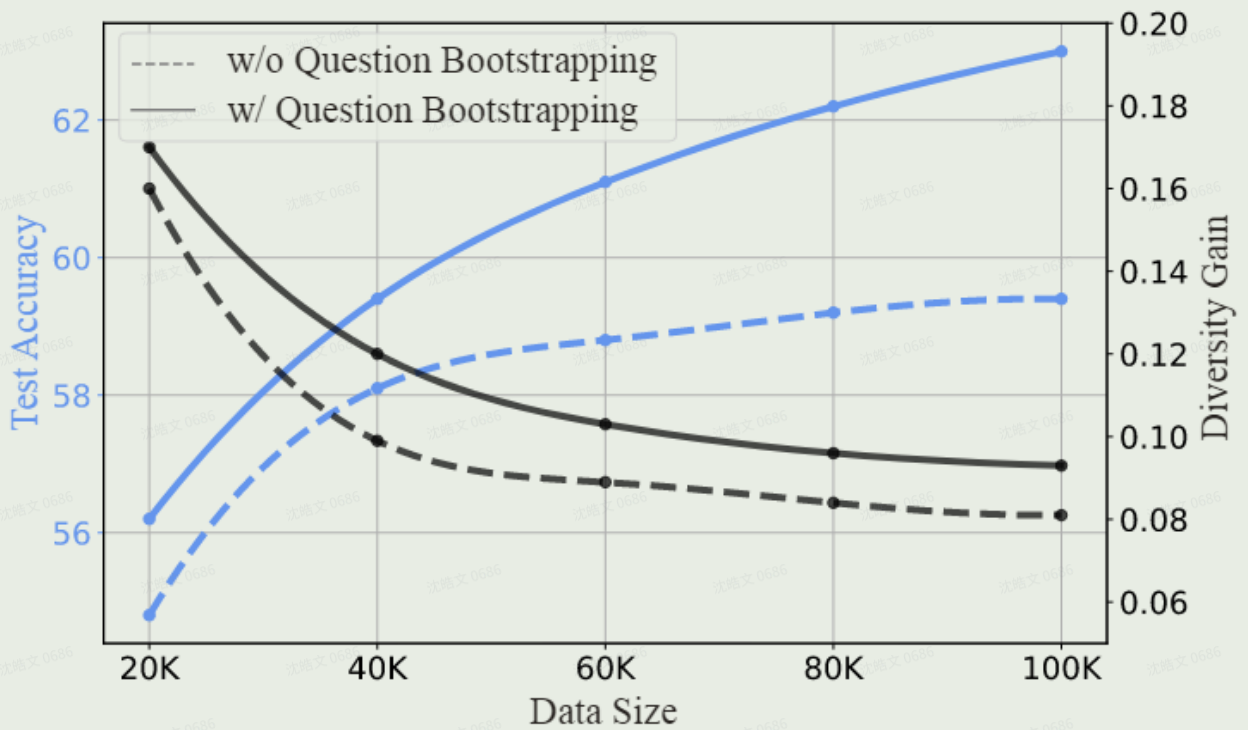


Figure 2: GSM8K accuracy of LLaMA-2-7B finetuned on different sizes of answer augmentation data. Larger diversity gain indicates the question is more diverse compared to the existing questions. Detailed experimental setup is given in Section 4.1.

- The test accuracy without bootstrap problems quickly reaches saturation. In contrast, when using bootstrap problems, the test accuracy continues to show stable improvement.
- MetaMathQA includes four methods of **data augmentation** :
 - i. **Answer Augmentation (Answer Augmentation):** Given a question, generate a thinking chain (reasoning path) that can obtain the correct result through a large language model as data augmentation.
 - Question: James buys 5 packs of beef that are 4 pounds each. The price of beef is \$5.50 per pound. How much did he pay?
 - Answer: James buys 5 packs of beef that are 4 pounds each, so he buys a total of $5 * 4 = 20$ pounds of beef. The price of beef is \$5.50 per pound, so he pays $20 * \$5.50 = \110 . The answer is: 110.
 - ii. **Rephrasing Question (Question Rewriting Enhancement):** Given a meta-question, rewrite the question through a large language model and generate a thought chain that obtains the correct result as data augmentation.

- Question: What is the total amount that James paid when he purchased 5 packs of beef, each weighing 4 pounds, at a price of \$5.50 per pound?
- Answer: Each pack of beef weighs 4 pounds, so 5 packs weigh $4 * 5 = 20$ pounds in total. The price per pound of beef is \$5.50, so the total cost for 20 pounds is $20 * \$5.50 = \110 . Therefore, James paid a total of \$110. The answer is: 110.

iii. **Self-Verification Question (Self-Verification Reverse Question Enhancement):**

Based on FOBAR, data augmentation is carried out by rewriting the statements in the reverse question section through a large language model. Questions with answers are first rewritten as declarative statements, and then an additional question asking for the value of x is added.

- Question: James buys x packs of beef that are 4 pounds each. The price of beef is \$5.50 per pound. He paid 110. What is the value of unknown variable x ?
- Answer: To solve this problem, we need to determine the value of x , which represents the number of packs of beef that James bought. Each pack of beef weighs 4 pounds and costs \$5.50 per pound. The total amount James paid is \$110. We can set up the equation as follows: Number of packs of beef * Weight per pack * Price per pound = Total amount paid; $x * 4 * \$5.50 = \110 ; Let's simplify and solve for x : $4x * \$5.50 = \110 ; $22x = \$110$; To isolate x , we divide both sides of the equation by 22: $22x / 22 = \$110 / 22$; $x = \$5$; The value of x is 5.

iv. **FOBAR Question (FOBAR Reverse Problem Enhancement):** Given a meta-question, generate an inverse question by masking the number x in the condition, giving the original answer and backtracking x , and generate the correct thinking chain process based on the inverse question to augment the data. The question with the answer needs to be rewritten as a declarative statement (backtracking example: "If we know the answer to the above question is 110, what is the value of unknown variable x ?").

- Question: James buys x packs of beef that are 4 pounds each. The price of beef is \$5.50 per pound. How much did he pay? If we know the answer to the above question is 110, what is the value of unknown variable x ?
- Answer: James buys x packs of beef that are 4 pounds each, so he buys a total of $4x$ pounds of beef. The price of beef is \$5.50 per pound, so the total cost of the beef is $5.50 * 4x = 22x$. We are given that the total cost is \$110, so we can write: $22x = 110$. Dividing both sides by 22, we get: $x = 5$. The value of x is 5.

v. We combined all the augmented data (including answer augmented data and lead questions (repetition, self-validation, FOBAR) into the **MetaMathQA dataset**, which we used to fine-tune LLAMA-2

- Based on LLaMA-2 fine-tuning on the MetaMathQA dataset, a large language model focusing on mathematical reasoning (forward and reverse) **MetaMath** is obtained, achieving state-of-art on the mathematical reasoning dataset

- Goodness function: MLE

3. Experiments

1. Open source model

Model	#params	GSM8K	MATH
<i>closed-source models</i>			
GPT-4	-	92.0	42.5
GPT-3.5-Turbo	-	80.8	34.1
<i>open-source models (1-10B)</i>			
LLaMA-2	7B	14.6	2.5
Falcon	7B	6.8	2.3
ChatGLM 2	6B	32.4	-
Qwen	7B	51.6	-
Baichuan-2	7B	24.5	5.6
SFT	7B	41.6	-
RFT	7B	50.3	-
WizardMath	7B	54.9	10.7
MetaMath	7B	66.5	19.8
<i>open-source models (11-50B)</i>			
LLaMA-2	13B	28.7	3.9
LLaMA-2	34B	42.2	6.2
Falcon	40B	19.6	2.5
Vicuna	13B	27.6	-
Baichuan-2	13B	52.8	10.1
SFT	13B	50.0	-

RFT	13B	54.8	-
WizardMath	13B	63.9	14.0
MetaMath	13B	72.3	22.4

open-source models (51-70B)

LLaMA-2	70B	56.8	13.5
RFT	70B	64.8	-
WizardMath	70B	81.6	22.7
MetaMath (by QLoRA)	70B	82.3	26.6

- Without external tools (e.g., code interpreter), MetaMath significantly outperforms existing **open source LLM models**

2. Why is MetaMathQA useful? It improves the quality of MindChain data (Perplexity).

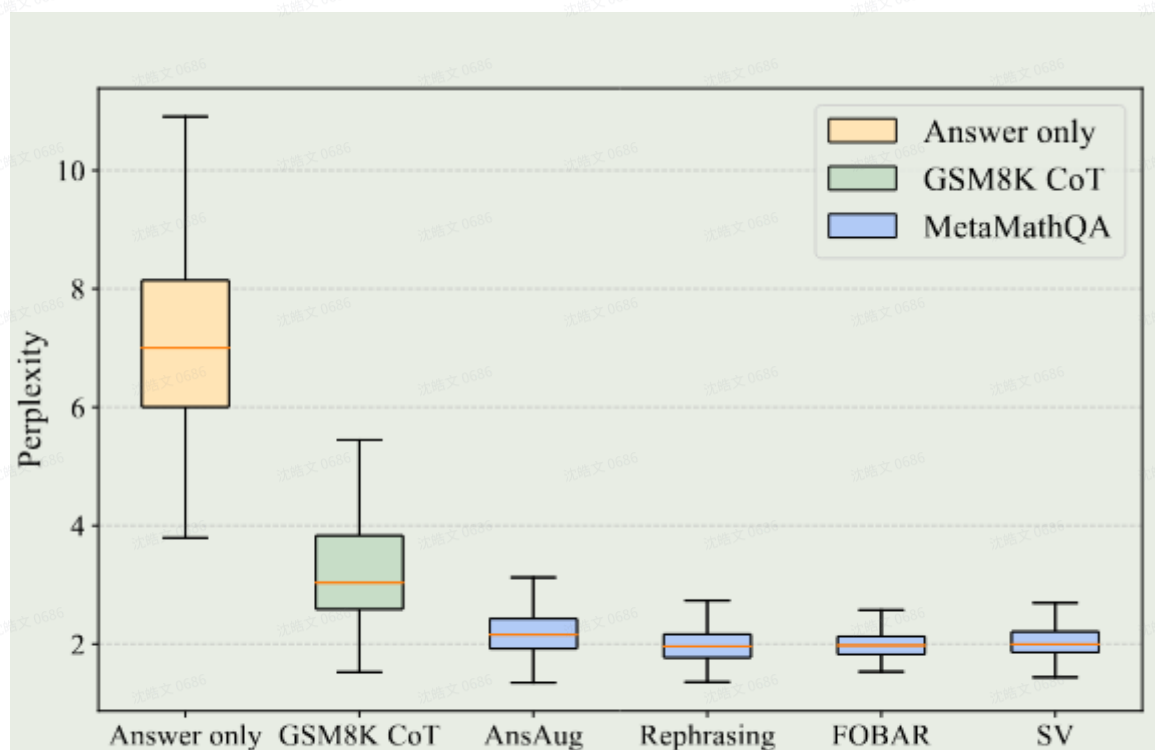
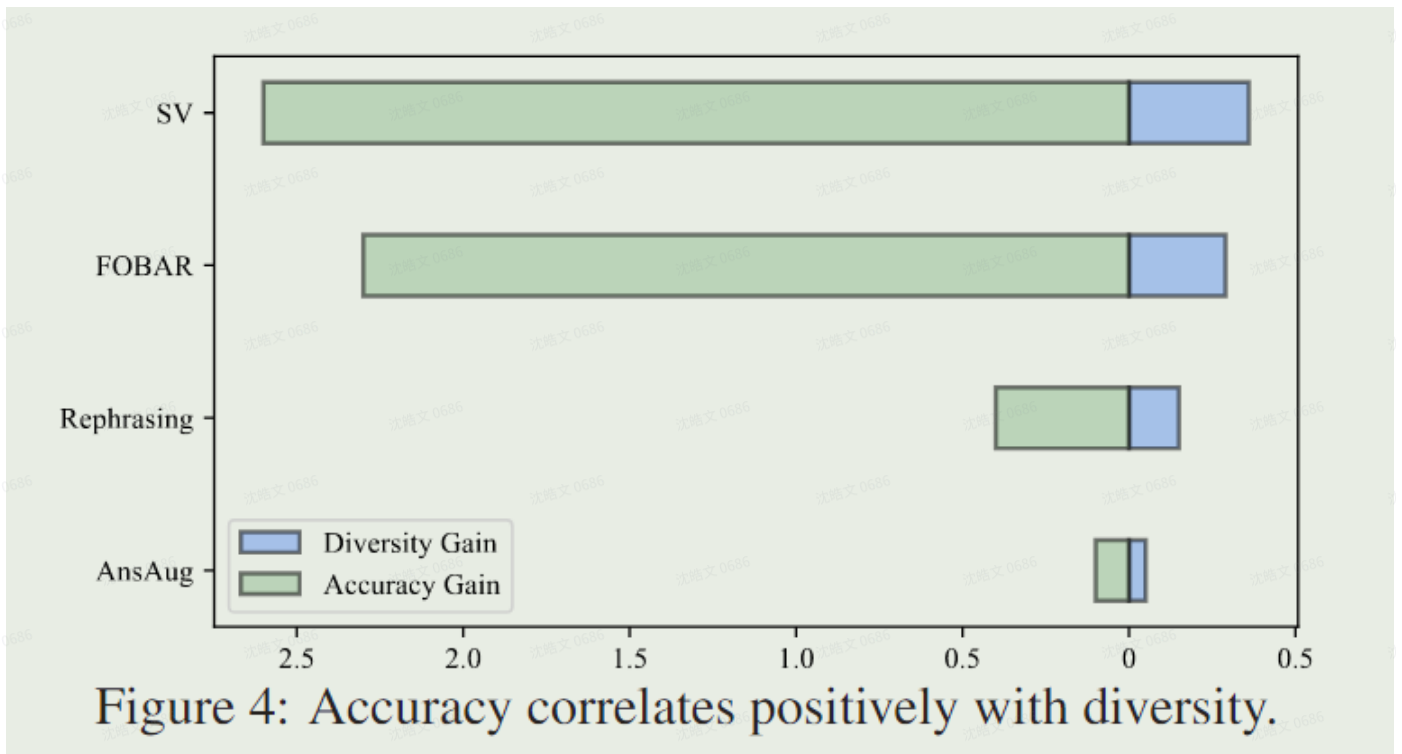


Figure 3: Lower perplexity of MetaMathQA.

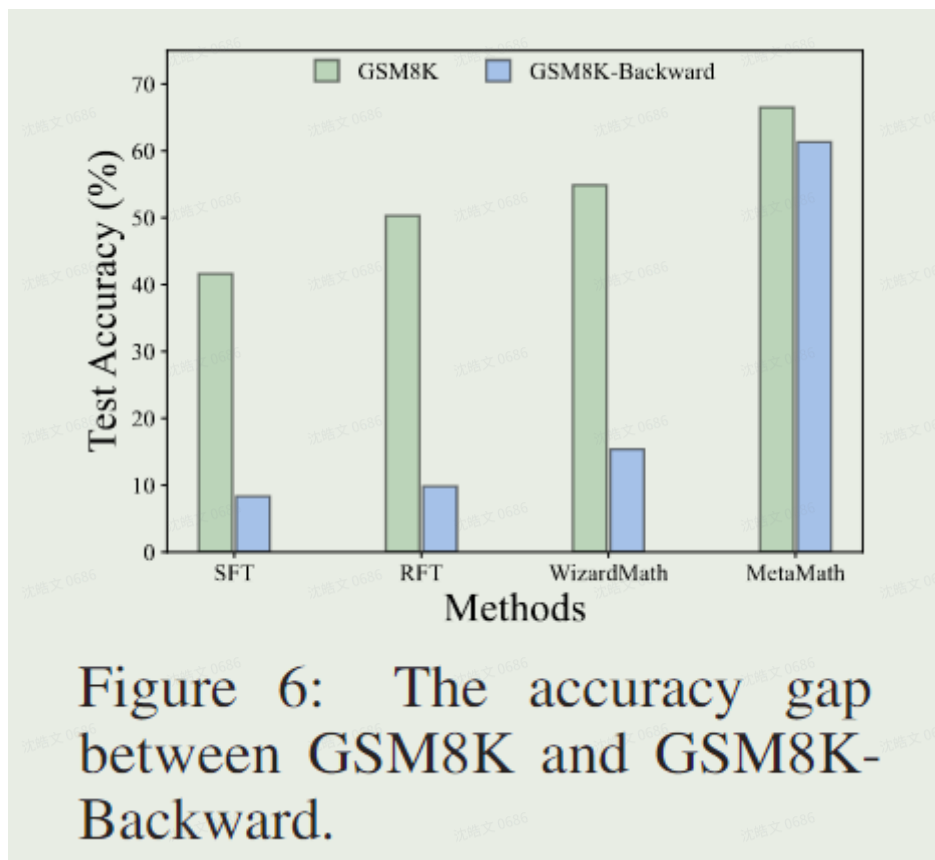
- The confusion of MetaMathQA is significantly lower than that of the other two datasets. This highlights its inherent easy-to-learn nature, which may be more conducive to deriving supported problem-solving capabilities from LLM

3. Why is MetaMathQA useful? It increases the diversity of thinking chain data.



- There is a positive correlation between the diversity and accuracy brought by the guidance method

4. Evaluate backward mathematical ability



- Proposed a GSM8K-Backward test set, including the use of SV and FOBAR (inverse problem enhancement method) to enhance the original GSM8K test set (1270 backward problems)

- The existing LLM is difficult to solve the mathematical problem of backward basic principles, and MetaMath has significant improvements on both datasets

Arithmo-Mistral-7B

1. Contributions

- Model Training Dataset: Combining MetaMathQA (training dataset), lila OOD (training dataset, validation set, and test set), and MathInstruct (training dataset) datasets
- We have verified that our training data does not overlap with the GSM8K and MATH test sets
- Apply further post-processing steps, such as
 - Deduplication
 - Randomly reduce the case of x% input
 - Add different set of Python hints for PoT
 - Standardized answer format
- The size of the final dataset is~ 540,000.
- Use QLoRA to fine tune it on a single RTX 4090 GPU

2. Experiments

Model	GSM8k Pass@1	MATH Pass@1
MPT-7B	6.8	3.0
Falcon-7B	6.8	2.3
LLaMA-1-7B	11.0	2.9
LLaMA-2-7B	14.6	2.5
MPT-30B	15.2	3.1
LLaMA-1-13B	17.8	3.9
GPT-Neo-2.7B	19.5	--
Falcon-40B	19.6	2.5
Baichuan-chat-13B	23.9	--
Vicuna-v1.3-13B	27.6	--
LLaMA-2-13B	28.7	3.9
InternLM-7B	31.2	--

ChatGLM-2-6B	32.4	--
GPT-J-6B	34.9	--
LLaMA-1-33B	35.6	3.9
LLaMA-2-34B	42.2	6.24
RFT-7B	50.3	--
LLaMA-1-65B	50.9	10.6
Qwen-7B	51.6	--
WizardMath-7B	54.9	10.7
LLaMA-2-70B	56.8	13.5
WizardMath-13B	63.9	14.0
MetaMath-7B	66.5	19.8
MetaMath-13B	72.3	22.4
🔥 Arithmo-Mistral-7B Zero-Shot PoT	71.2	--
🔥 Arithmo-Mistral-7B Zero-Shot CoT	74.7	25.3
WizardMath-70B	81.6	22.7
MetaMath-70B	82.3	26.6

- The benchmark scores on these two datasets are very high

Abel

- We show that:
 - **without** tools
 - **without** continuing pretraining
 - **without** reward model
 - **without** RLHF
 - **ONLY** using SFT

can establish a new **state-of-the-art** performance across open-source LLMs on the GSM8k (83.62) and MATH (28.26) benchmarks

1. Contributions

- **Parental Oversight:** A Babysitting Strategy for **Supervised Fine-tuning**

- The processing method of fine-tuning data significantly affects the performance of trained GAI
- This principle emphasizes cautious handling of supervisory fine-tuning
- Similar to encouraging parents to educate their children. Different types of data and their presentation formats (such as step-by-step reasoning, iterative refinement) can be compared to different educational methods. Just as parents carefully choose the most effective method to guide their children, GAI practitioners should also carefully choose the most effective data processing method to better guide their LLM
- The idea that "more data is better" is not always true. **The quality and relevance of annotated samples often exceed their quantity**
- The training samples used in SFT should not only provide the correct answers, but also guide the model on how to derive the correct answers based on LLM knowledge
- If LLM knowledge is insufficient to answer questions, timely intervention should be made to address knowledge gaps

2. Experiments

Model	GSM8k	MATH	MathQA	SVAMP	SCQ5K-EN	ARC-E	ARC-C	HellaSwag	MMLU
Abel-7B-002	80.44	29.46	69.78	77.67	55.95	77.67	55.05	77.72	61.19
Abel-7B-001	59.74	13	1.21	57.67	9.3	53.32	38.97	63.51	40.59
MetaMath-Mistral-7B	77.7	28.2	33.94	79.33	37.6	78.48	51.93	76.44	61.93
Qwen-7b	47.84	9.34	27.44	53	40.05	74.97	53.05	86.85	57.98
Mistral-7b	37.83	9.06	25.73	63	39.6	76.83	53.22	76.31	64.05
Yi-6b	32.6	5.78	26.98	55.67	35.5	73.66	49.53	68.97	64.02
LLaMA2-7b	12.96	2.78	11.52	44	28.24	71.12	46.61	71.32	46.7

- Abel-002 performs well on mathematical datasets (GSM8K, MATH, MathQA, SVAMP, SCQ5K-EN)
- It is also competitive on out-of-domain inference datasets (ARC-E, ARC-C, HellaSwag), surpassing the basic model Mistral-7b
- On MMLU, Abel-7B-002 shows only 3 points, compared to mistral-7b, Abel-7B-001 shows a decrease of 6 points, and compared to LLaMA2-7b, it shows a decrease of 6 points.
- We implemented **state-of-art** across open source LLM (without using external tools) on GSM8k (**83.62**) and MATH (**28.26**)

- GSM8K's performance is **83.62** , surpassing top models such as PaLM-1, Minerva (Google), Claude-instant (Anthropic), ChatGPT (OpenAI), and only lagging behind Google's latest model PaLM-2-Flan by 1 percentage point.
- On **a highly challenging math competition problem** , the accuracy rate was 28.26% (42.5% for GPT4), significantly ahead of other open source models, and 5.46% higher than the previous best open source model.
- Using our approach, we not only achieved excellent results on GSM8K and MATH, but when given **a new dataset** (TALSCQ-EN), we quickly achieved state-of-the-art (SOTA) performance, effortlessly surpassing the multi-billion dollar business models MathGPT and GPT4

RFT & Factors influencing mathematical reasoning ability of Supervised LLM

1. Motivation

When using SFT, a **logarithmic linear relationship** is found between data volume and model performance: as the supervised dataset increases, the model performance improves less

2. Contributions

1. Propose RFT: Rejection Sampling Fine-tuning

- Generate k different reasoning paths through the model fine-tuned by SFT
- Delete content with inconsistent answers
- Each inference path contains a list of equations, and one inference path is selected as the enhanced data for each different list of equations
- Obtain enhanced dataset: $D'_\pi = D \cup \{q_i, r_{ij}, a_{ij}, i_{ij}\}$
- Fine-tune large models with augmented datasets

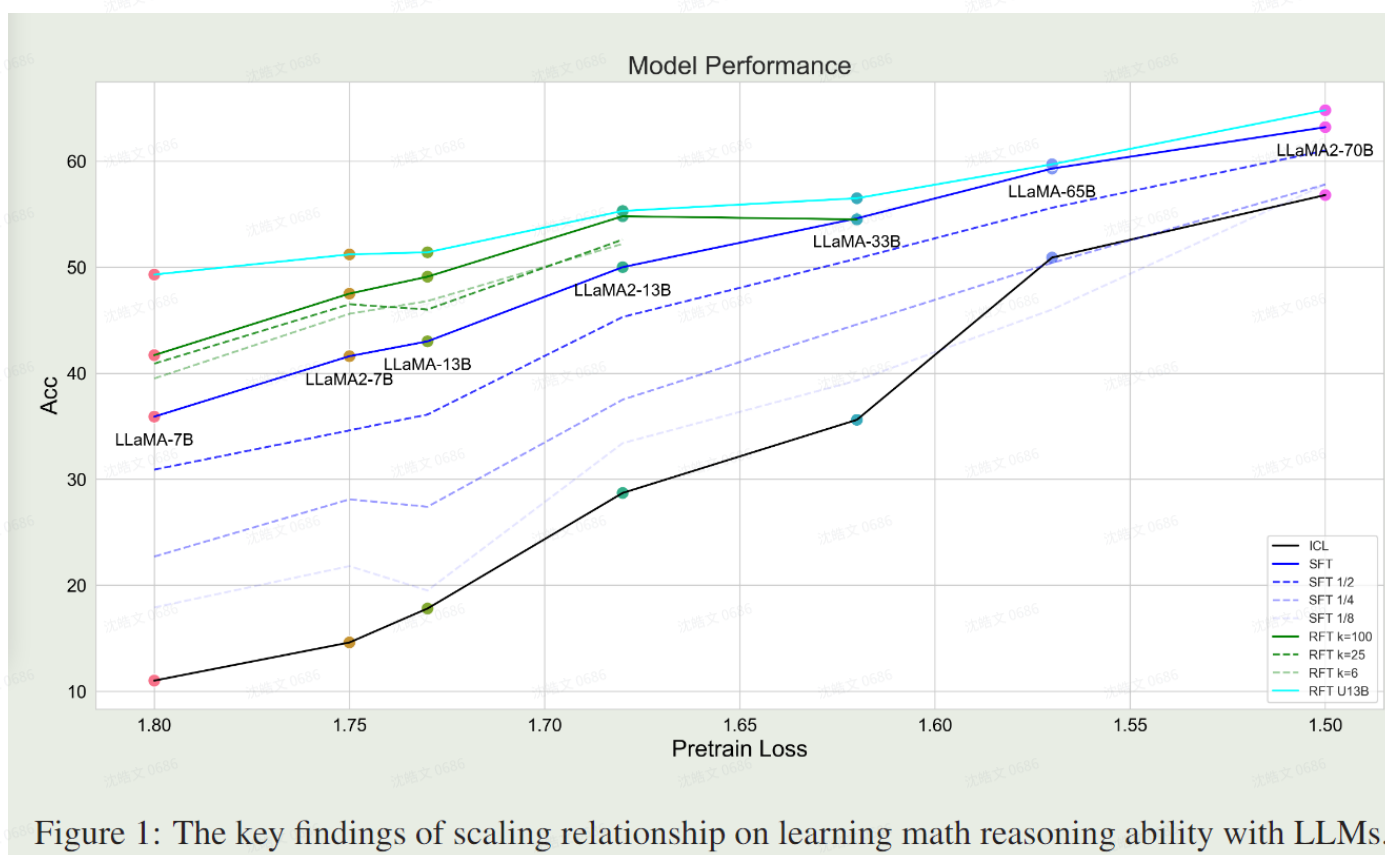
2. This paper empirically studies the **factors affecting the mathematical reasoning ability of Supervised LLM** , including **Pre-training loss** , **supervised data volume** and **enhanced data volume**

- Within a given interval, the training loss is approximately negatively correlated with SFT and ICL accuracy
- The performance of the model is logarithmically linear with the amount of supervised data, and this performance improvement shrinks as the training model improves

- c. Hope to use the model itself to generate more supervised data to enhance its inference ability and analyze the scaling relationship of enhanced data volume. Apply rejection sampling to the SFT model for sampling and select the correct inference path as the enhanced dataset. We use these enhanced datasets to fine-tune the basic LLM, which will achieve better performance compared to SFT, and we represent it as rejection sampling fine-tuning (RFT).
 - i. We discussed the reason why RFT works is that it provides multiple inference paths, which makes LLM have better inference generalization.
 - ii. We also discussed the basic solution that RFT is much cheaper than pre-training in computational resources when training LLM with lower pre-training loss.

3. Findings

- o RFT
 - RFT uses supervised models to generate and collect correct inference paths as an enhanced fine-tuning dataset
 - RFT improves the mathematical inference performance of LLM
 - RFT brings more improvements to LLM with lower performance
- o Factors Affecting the Mathematical Reasoning Ability of Supervised LLM

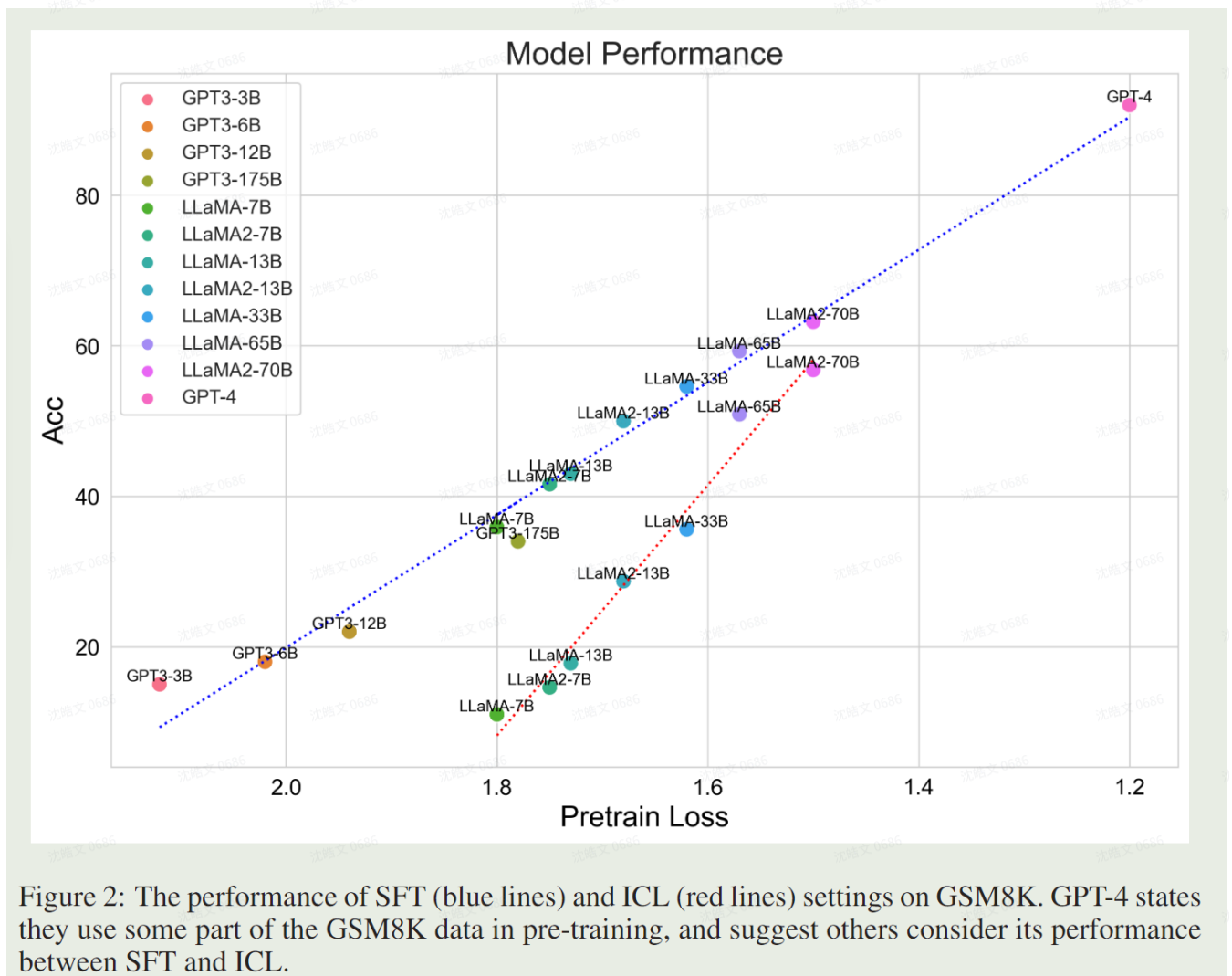


- a. When the pre-training loss is smaller (i.e. the pre-trained model is better), the inference performance of SFT and ICL models increases linearly within a certain range. SFT performance improves slower than ICL.

- b. SFT increases logarithmically with the increase of supervised data volume. As the pre-trained model becomes better, the benefits of increasing data volume will decrease.
- c. The model performance of RFT improves with the increase of the number of different inference paths, and the performance improvement of RFT is slower than that of SFT.
- d. The combination of rejected samples from multiple models further enhances the RFT performance.

3. Experiments

1. Factors affecting the mathematical reasoning ability of supervised large language models
 pre-trained LLM $r \rightarrow (D = \{q_i, r_i, a_i\}_i) \rightarrow$ SFT model $\pi \rightarrow$ (greedy decoding) \rightarrow generate **reasoning paths** and **answer** $>$ Accuracy
 - a. Model accuracy vs. pre-training loss



- **Pre-training loss** is a stable performance indicator of mathematical inference ability and should be used to represent models rather than their model parameters and number of pre-training tokens
- The result of fine-tuning on gsm8k

- Within the given pre-training loss interval, the pre-training loss is approximately negatively correlated with SFT and ICL accuracy
- SFT (blue line) is always better than ICL (red line), and when the loss is low before training, the improvement will decrease
- From observation, an effective way to improve inference is **to train better base models with lower pre-training loss**

b. Model accuracy vs. amount of supervised data

Fine-tune LLAMA and LLAMA2 with {1, 1/2, 1/4, 1/8, 1/16, 1/32} quantities from the training dataset from GSM8K

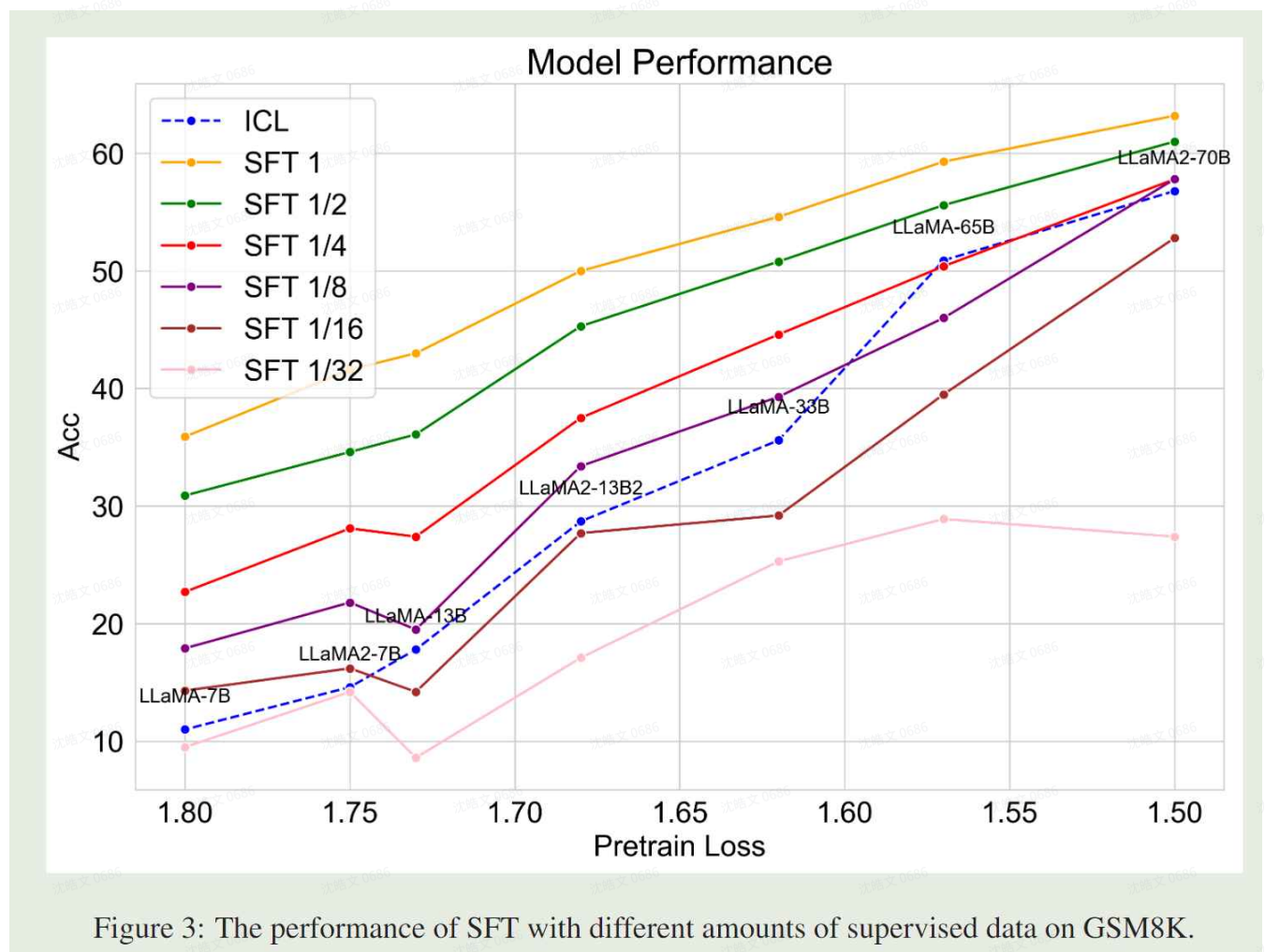


Figure 3: The performance of SFT with different amounts of supervised data on GSM8K.

- Model performance and data volume have a **logarithmic linear relationship**
- Good models require more data to outperform their ICL performance
- When the amount of supervised data doubles, better models benefit less: better models learn more inference ability in pre-training

c. Model accuracy vs. Enhanced data counting

The key factor affecting RFT augmented data fine-tuning is the **number of different inference paths**

- i. Results of RFTs **sampled with $k = 100$ candidate inference paths** on LLaMA and LLaMA-2 (temperate = 0.7)

Setting	7B	7B-2	13B	13B-2	33B
Pretrain loss	1.8	1.75	1.73	1.68	1.62
ICL	11.0/18.1	14.6/-	17.8/29.3	28.7/-	35.6/53.1
SFT	35.9/48.7	41.6/55.4	43.0/55.2	50.0/61.7	54.6/-
RFT $k = 100$	41.7/52.7	47.5/58.7	49.1/59.9	54.8/65.4	54.5/-
Correct paths per question	53.3	60.8	62.5	71.6	88.7
Distinct paths per question	5.25	5.19	5.26	5.29	2.78

Table 1: The performance of RFT with $k = 100$ on GSM8K compared with SFT and ICL. Distinct path amount means distinct equation list amount here.

- For the 33B model, RFT does not improve performance compared to SFT: it is difficult to generate diverse inference paths, and using larger temperate effects is still not good
- ii. Performance of RFTs with different numbers of sample counts k on GSM8K

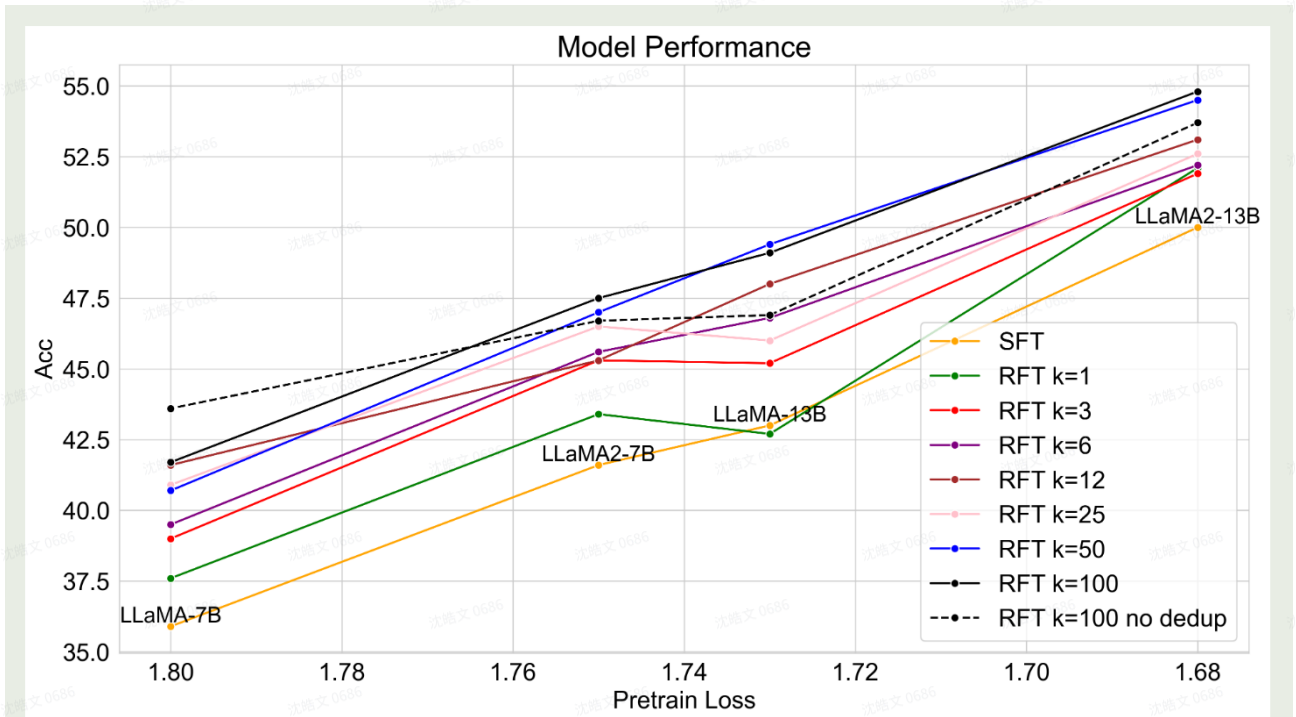


Figure 4: The performance of RFT with different amounts of sampling count k on GSM8K.

- Estimating RFT performance based on different **number of inference paths** is better than based on number of samples
- $K = 100$ has similar performance compared to RFTs without dedup (without deleting the reasoning path)
- Using deduplication has better performance for 3 out of 4 models and requires less training time
- When $k = 3$, RFT stability is better than SFT

- For most Data Points, using a larger k value will bring better performance. However, when k doubles, the performance improvement brought by RFT will be weakened and reduced

k	7B	7B-2	13B	13B-2	33B
1	1.17	1.19	1.15	1.18	1.06
3	1.44	1.47	1.41	1.45	1.16
6	1.74	1.78	1.69	1.76	1.28
12	2.20	2.23	2.11	2.21	1.46
25	2.93	2.93	2.88	2.94	1.77
50	3.94	3.91	3.90	3.94	2.19
100	5.25	5.19	5.26	5.29	2.78
400 (U13B)				12.84	
500 (U33B)				13.65	

Table 2: Different reasoning paths per question generated by different SFT models with different k .

iii. Performance of RFT combining multiple models

$$D'_{U13B} = D'_{7B} \oplus D'_{7B2} \oplus D'_{13B} \oplus D'_{13B2} \quad \text{and} \quad D'_{U33B} = D'_{U13B} \oplus D'_{33B}$$

- It represents an aggregation process. First, all inference paths from different sets are combined together, and then algorithm 1 (shown in the figure below) is applied to deduplicate the inference paths with the same calculation process and order

Algorithm 1: Reasoning Path Selection

Data: Reasoning paths for question q , \mathcal{R}_q

Result: Selected reasoning paths for question q , \mathcal{R}_q^s

```

1 Initialize selected reasoning paths,  $\mathcal{R}_q^s = \text{list}()$ 
2 Initialize appeared equation set,  $\mathcal{E}_q^s = \text{set}()$ 
3 for  $r$  in  $\mathcal{R}_q$  do
4     if  $\text{get\_equation}(r) \notin \mathcal{E}_q^s$  then
5          $\mathcal{R}_q^s.\text{append}(r)$ ;
6          $\mathcal{E}_q^s.\text{update}([\text{get\_equation}(r)])$ 
7     end
8     else
9         find  $r^s \in \mathcal{R}_q^s$  s.t.  $\text{get\_equation}(r^s) = \text{get\_equation}(r)$ ;
10        if  $\sum_{i:r_i^s \in \mathcal{E}_q^s, r_i^s \neq r^s} \text{Levenstein\_dist}(r, r_i^s) > \sum_{i:r_i^s \in \mathcal{E}_q^s, r_i^s \neq r^s} \text{Levenstein\_dis}$ 
11             $r^s = r$ ;
12        end
13    end
14 end

```

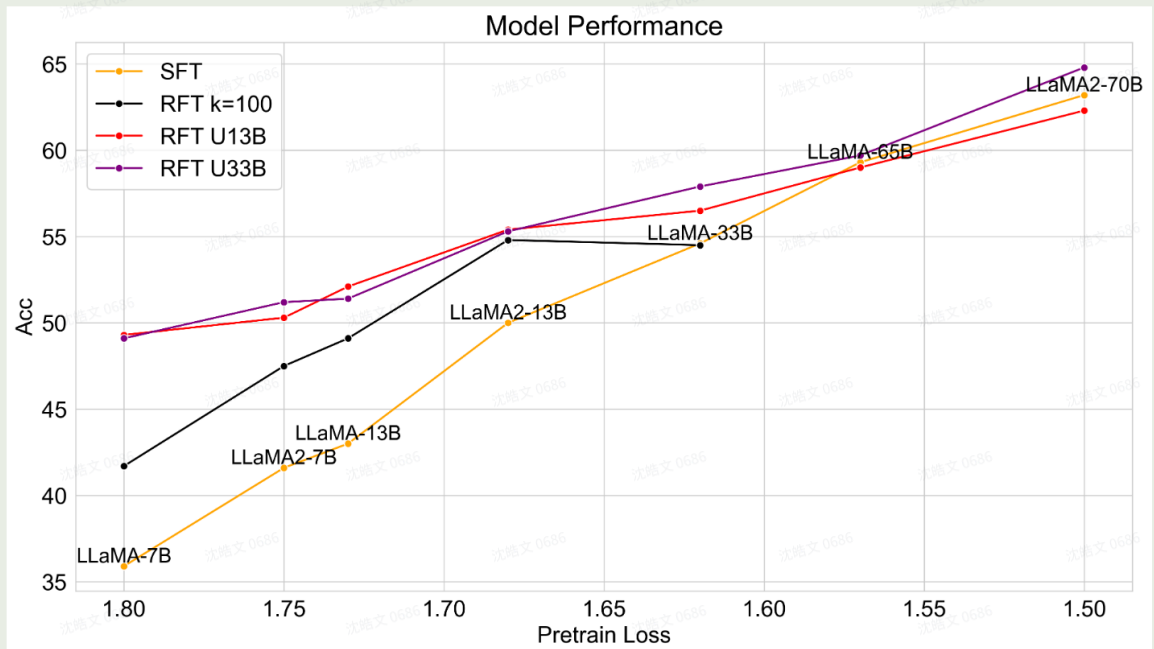


Figure 5: The performance of RFT with rejection sampling samples from multiple models.

- The combined enhanced dataset provides sufficient inference supervision to bridge the pre-training gap
- Applying RFT $k = 100$ on the 33B model is expensive, compared to sampling 100 times on the 33B, D'_{U13B} fine-tuning on has similar rejection sampling computational costs and can achieve better performance
- Inclusion in the aggregation D'_{U33B} has almost no impact on performance: the SFT model of 33B can only **provide limited inference diversity** when sampling the training problem, indicating that the 33B model (and possibly the 65B and 70B models) can **remember the inference path annotated manually**
- For the 65B model, we found that compared to SFT, using D'_{U13B} did not improve performance. The reason may be that better models benefit less from supervised sample size, while **learning more inference ability in pre-training**

Base Model	Training	maj1@1	maj1@K*
Proprietary LLMs			
GPT-4 (OpenAI, 2023)	5-shot ICL	92.0	-
GPT-3-175B (Brown et al., 2020)	SFT	34.0	-
PaLM2 (Anil et al., 2023)	8-shot ICL	80.7	91.0@K=40
PaLM-540B (Chowdhery et al., 2022)	8-shot ICL	56.5	74.4@K=40
Chinchilla-70B (Uesato et al., 2022)	5-shot ICL	43.7	58.6@K=96
Chinchilla-70B	SFT	58.9	77.7@K=96
Open-sourced LLMs			
GPT-Neo-2.7B (Black et al., 2021)	FCS + PCS (Ni et al., 2023)	19.5	41.4
GPT-J-6B (Wang & Komatsuzaki, 2021)	CoRE (Zhu et al., 2023)	34.9	63.2@K=40
ChatGLM2-6B (Zeng et al., 2022)	8-shot ICL	32.4	-
ChatGLM2-6B	Human Alignment	28.1	-
ChatGLM2-12B	8-shot ICL	40.9	-
ChatGLM2-12B	Human Alignment	38.1	-
InternLM-7B (Team, 2023)	4-shot ICL	31.2	-
InternLM-7B	Human Alignment	34.5	-
LLaMA-7B	SFT	35.9	48.7
Our RFT on open-sourced LLMs			
LLaMA-7B	RFT-U13B	49.3	61.8
LLaMA2-7B	RFT-U13B	50.3	65.6
LLaMA-13B	RFT-U13B	52.1	66.2
LLaMA2-13B	RFT-U13B	55.4	69.1

Table 3: Compare GSM8K results with other baselines. RFT-U13B means models fine-tuned on \mathcal{D}'_{U13B} . FCS and PCS represent fully-correct solutions and partially-correct solutions respectively. *K=100 if not specified.

(maj1@k means we generate k samples for each question and do a majority vote)

MathPile mathematical pre-training dataset (homologous to Abel)

1. Contributions

- Purpose: **Quality >> Quantity**
- Features

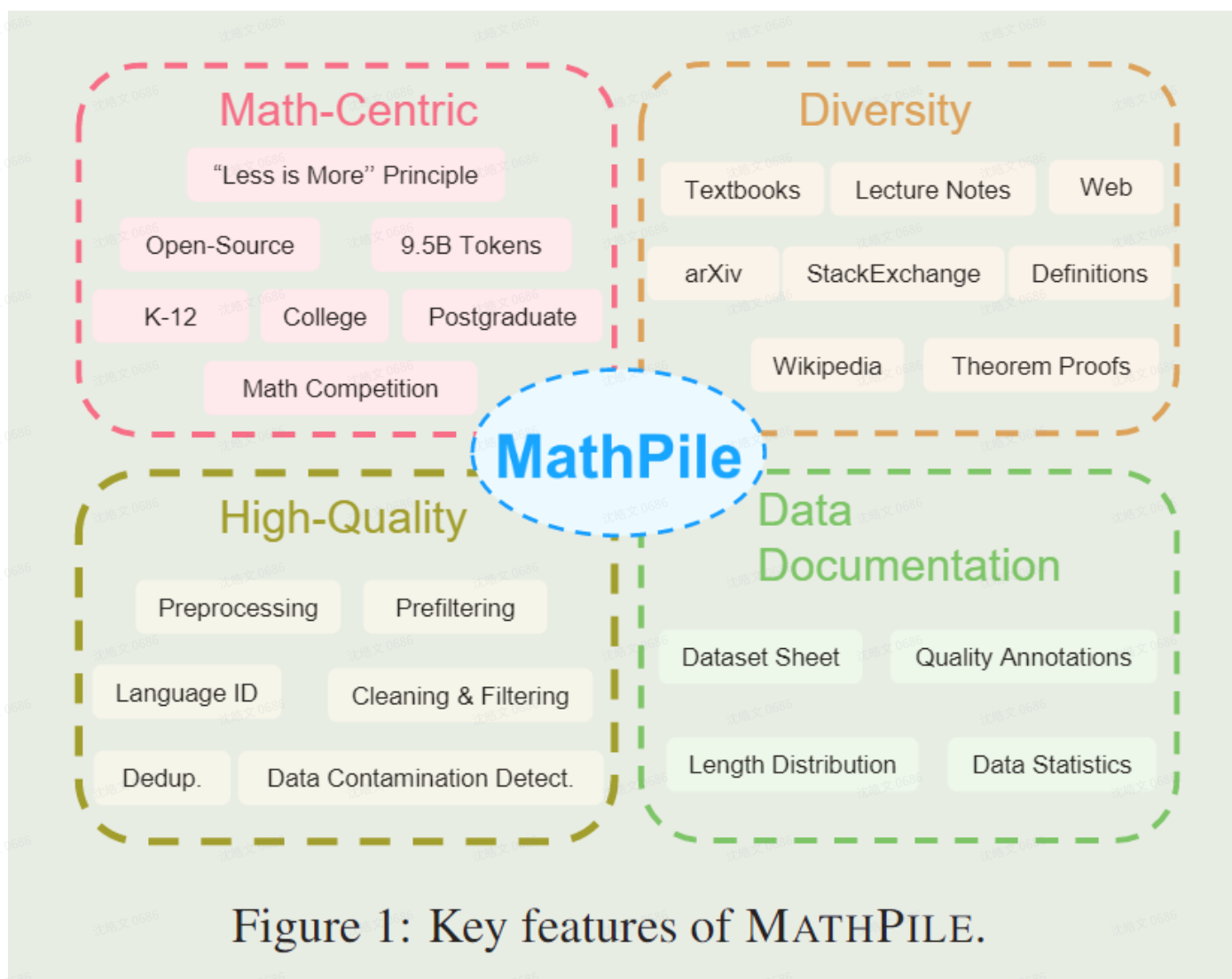


Figure 1: Key features of MATHPILE.

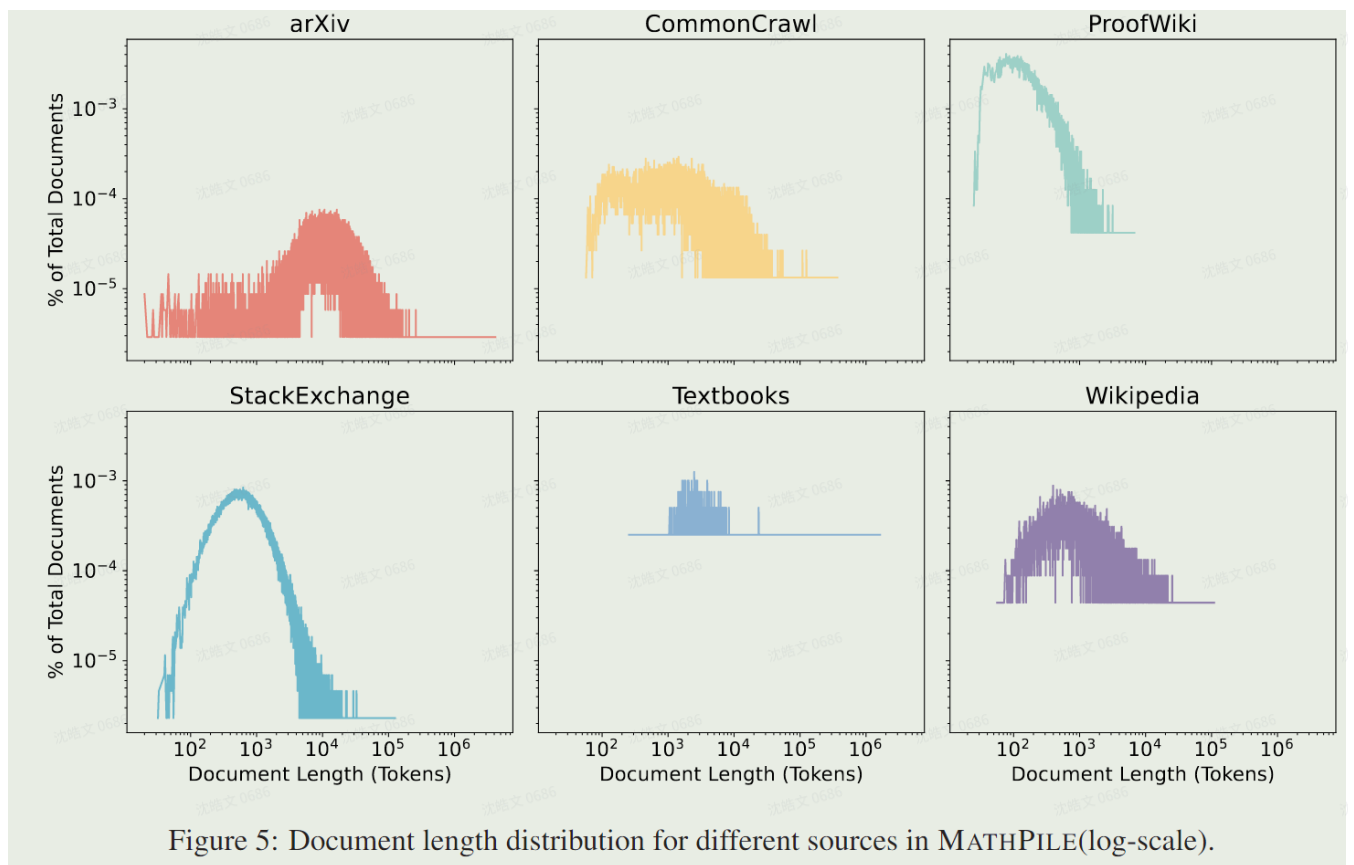
- Math-centric: Previous open-source pre-training corpora have typically focused on general domains or focused on multiple languages or programming languages, lacking **corpora specifically tailored for mathematics**
- Diversity: Our corpus goes beyond the web, integrating **high-quality mathematics textbooks, lecture notes, scientific papers in the field of mathematics from arXiv**, and carefully selected content from **StackExchange, ProofWiki, and Wikipedia**, making our corpus a **richer and more diverse** mathematical resource for language models
- High-Quality: Adverse effects of low-quality and duplicate content in pre-training corpus on Model Training. To achieve high-quality corpus, we have carried out **extensive preprocessing, pre-filtering, cleaning, filtering, and deduplication work**

- Data Documentation: For large-scale pre-training corpora, it is very important to **record the characteristics, expected uses, information content, and potential biases of the data in detail**. In the processing flow of this dataset, a large number of document annotations were carried out, such as language recognition scores and symbol-to-vocabulary ratios. These quality annotations allow future users to apply specific filters based on these scores. The authors conducted extensive deduplicate work and performed downstream benchmark test set data pollution detection on the dataset, removing any identified duplicate samples.
- Methods for collecting data
 - Mathematical Textbooks
 - Mathematical Papers from ArXiv
 - Mathematical Entries in Wikipedia
 - Entries from ProofWiki
 - Mathematical Discussions on StackExchange
 - Mathematical Web Pages from Common Crawl
- Data processing methods
 - Language Identification
 - Data Cleaning and Filtering
 - Data Deduplication
 - Data Contamination Detection
- Data Analysis
 - Overview

Components	Size (MB)	# Documents	# Tokens	max(# Tokens)	min (# Tokens)	ave (# Tokens)
Textbooks	644	3,979	187,194,060	1,634,015	256	47,046
Wikipedia	274	22,639	78,222,986	109,282	56	3,455
ProofWiki	23	23,839	7,608,526	6,762	25	319
CommonCrawl	2,560	75,142	615,371,126	367,558	57	8,189
StackExchange	1331	433,751	253,021,062	125,475	28	583
arXiv	24,576	343,830	8,324,324,917	4,156,454	20	24,211
Total	29,408	903,180	9,465,742,677	-	-	10,480

Table 4: The components and data statistics of MATHPILE.

- The Length Distribution of Documents



Thinking

1. Can the gap between open-source and closed-source models in Math Reasoning be filled by prompting or fine-tuning? (RFT: The smaller the pre-training loss, the smaller the improvement brought by this method.)

2.