Advancing Math Reasoning in Language Models: The Impact of Problem-Solving Data, Data Synthesis Methods, and Training Stages

Zui Chen · Tianqiao Liu · Tongqing · Mi Tian · Weiqi Luo · Zitao Liu

Introduction

CPT in math reasoning

Current Status

- Approach: Large-scale math-related corpus training (next-token prediction) to enhance mathematical reasoning.
- Key Issue: Data-intensive methods show low efficiency; limited improvement compared to supervised fine-tuning.
- Current Research: Focus on high-quality math data collection, but fails to significantly boost reasoning capabilities.

Key Hypotheses

Hypothesis 1: Knowledge Application > Knowledge Memorization

- Current methods: Filter math corpora to improve memorization, but yield marginal gains in reasoning.
- Proposed shift: Focus on teaching models how to use knowledge, not just expanding knowledge storage.

Hypothesis 2: The Unique Nature of Mathematical Knowledge

- Math knowledge is highly structured (axioms/theorems as core); the challenge lies in organizing and applying it to complex reasoning.
- Traditional memorization-based pretraining is inadequate for mathematical reasoning demands.

Providing **problem-solving data** corresponding to reasoning tasks is potentially more efficient!

Training data — Math corpus

Bayes and his Theorem
What Bayes' did was derive the special case of this formula for "inverting" the binomial distribution. This distribution gives the probability of x successes in n independent "trials" each having the same probability of success, p; each "trial" has only two possible outcomes ("success" or "failure"). Trials like this are usually called Bernoulli trials, after Daniel Bernoulli. If we ask the question "what is the probability of exactly x successes from the possible n?" ...

Knowledge Storage and Extraction

Target ability

Q: What is Bayes' theorem?
A: Bayes' theorem is a core theorem in probability theory, used to describe how the conditional probability of an event occurring updates under certain known conditions.

Training data — Problem-solving data

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: The answer is 11.
Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?
A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9.

Knowledge Storage and Extraction

Target ability

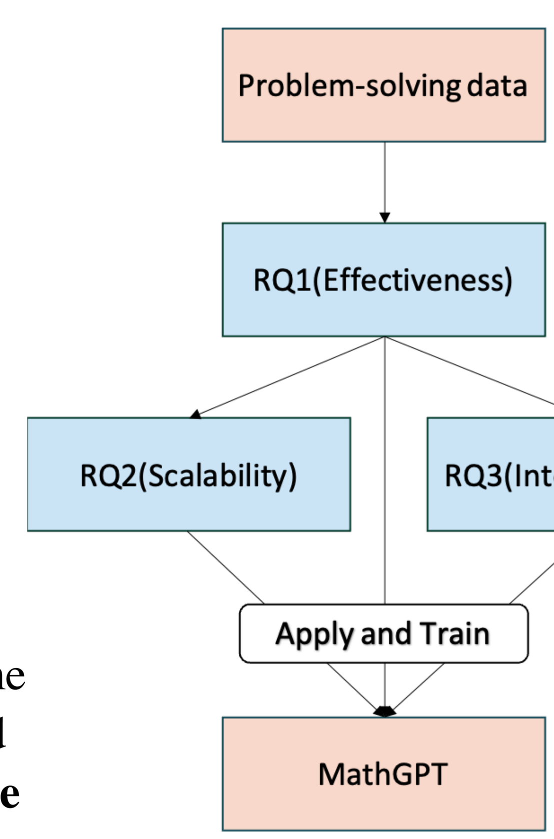
Q: Emily has 7 marbles. She buys 4 bags of marbles, and each bag contains 5 marbles. How many marbles does she have in total now?
A: Emily originally has 7 marbles. She buys 4 bags, each with 5 marbles, so she adds **4 × 5 = 20** marbles. **7 + 20 = 27**. Therefore, the answer is **27**.

Research Questions

RQ1: During the CPT stage, can **providing problem-solving data more effectively** enhance the model's math reasoning ability compared to using general math corpora?

RQ2: Are synthetic data from the same source equally effective, and **what synthesis methods are most efficient?**

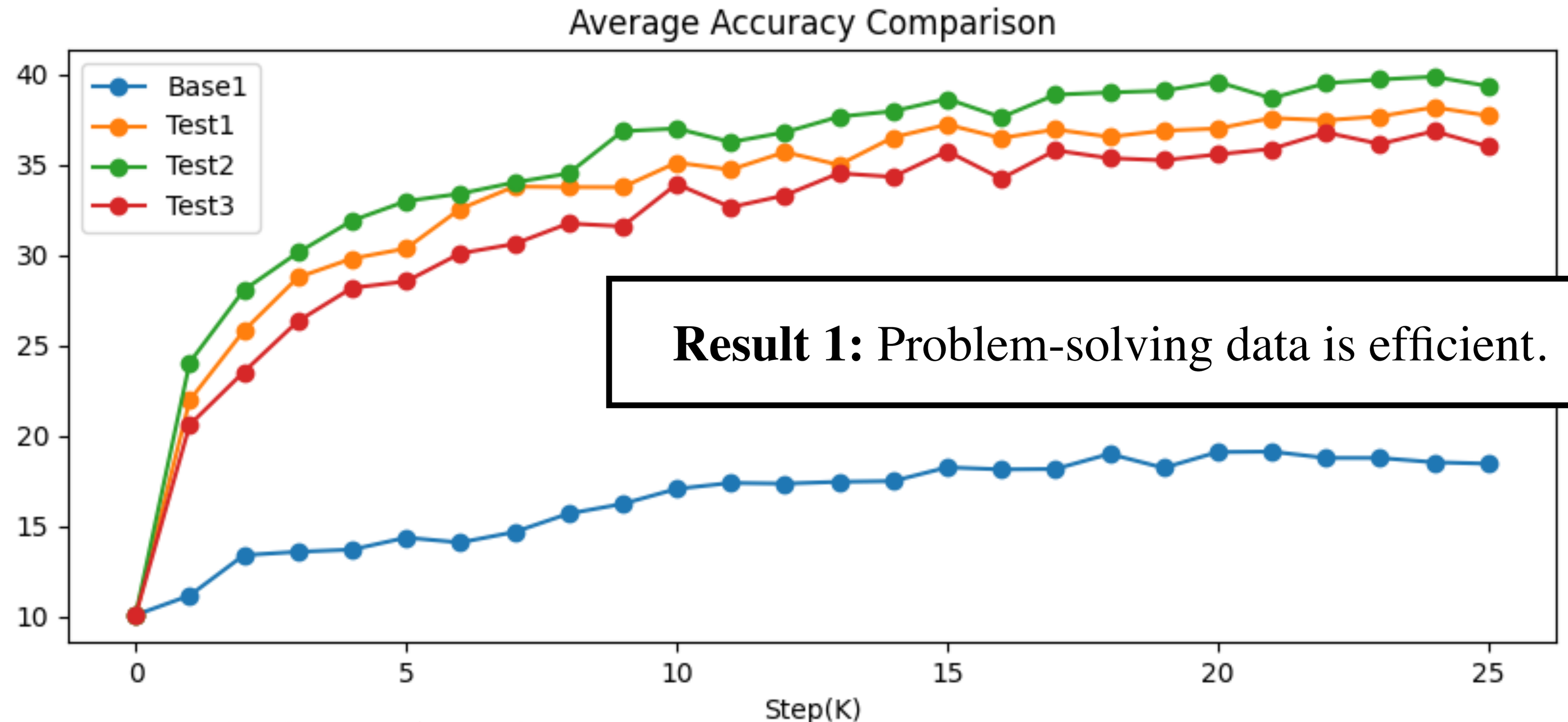
RQ3: How do the ability developed from the same problem-solving data differ between the CPT and SFT stages, and **what factors contribute to these differences**?



Experiments

Research Question 1-Effectiveness of problem-solving data

- Base1: Using 48.3B general corpus and 14.7B math corpus.
- Test1-3: Same to Base1, but **replacing math corpus with problem-solving data.**
- Test1, Test2, Test3 (math corpus : problem-solving data) = 3:7, 7:3, 5:5



Result 1: Problem-solving data is efficient.

Research Question 2-Synthesis methods of problem-solving data

Response Diversification
-> Increase response diversity
Input: Questions, Solution + Request to provide two different solutions
Output: <response> accept <\response> , Solution2, Solution3

Retrospective Enhancement
-> Insert subsequent step to make model learn to regret
Input: Questions, Solution($\text{step}_1, \ldots, \text{step}_i, \text{step}_{i+1}, \ldots, \text{step}_n$)
Insert: $\text{step}_1, \ldots, \text{step}_i, \langle\text{back}\rangle\text{step}_j\langle/\text{back}\rangle, \ldots, \text{step}_n \quad (i < j)$

Query Expansion -> Expand different queries
Input: Questions, Solution + Request to provide different questions and their solutions
Output: <statement> STATEMENT <\statement> , New question, New solution2, <check> accept <\check>

Retrospective Enhancement -> Imitate the real-life practice of teachers guiding students to rectify mistakes.
Input: Questions, Require students (small model) to answer and ask teachers (big model) to correct
Output: Student's Response + Teachers' Rectification

Model	Num	Tokens	GSM8K	MATH	GAOKAO	ZHONGKAO	Average
Base2	-	-	47.84	20.12	22.98	67.05	39.50
Res-Div	14,018,544	6.82B	52.99	23.22	23.83	65.15	41.30
Query-Exp	24,459,192	4.78B	51.25	23.08	27.23	69.13	42.67
Retro-Enh	14,707,792	5.04B	45.11	21.72	22.98	66.67	39.12
Tutor-Amp	11,942,328	13.90B	64.44	35.88	32.77	69.32	50.60

Result 2: Response Diversification, Query Expansion and Tutorship Amplification emerge as effective data synthesis techniques, with Tutorship Amplification registering particularly pronounced effects.

Research Question 3-Interpretability of ability acquisition difference between the CPT and SFT stages

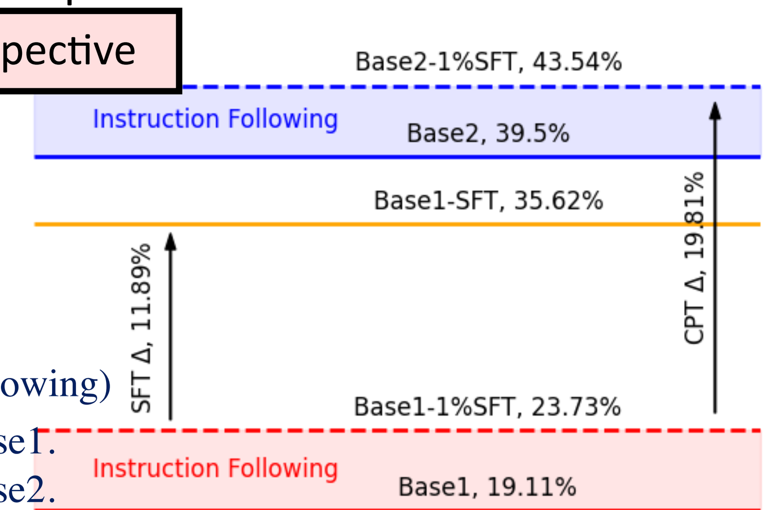
RQ3.1-Overall perspective

(Base2☺ vs Base1-SFT, same data in CPT and SFT)

- Base1: CPT with 48.3B general corpus and 14.7B math corpus.
- Base2: CPT with Base1 data and 7.2B problem-solving data.
- Base1-SFT: SFT with 7.2B problem-solving data based on Base1.

(Base2-1%SFT☺ vs Base1-1%SFT, ablation ability of instruction following)

- Base1-1%SFT: SFT with 0.072B problem-solving data based on Base1.
- Base2-1%SFT: SFT with 0.072B problem-solving data based on Base2.



Result 3: SFT has a clear disadvantage.

RQ3.2-Data distribution perspective

(CPT ☺ vs SFT on middle school data)

- Base1: CPT with 48.3B general corpus and 14.7B math corpus.
- Middle-school-SFT: SFT with 0.83B middle school data on Base1.
- Middle-school-CPT: CPT with Base1 data and middle school data.

(CPT ☺ vs SFT on high school data)

- High-school-SFT: SFT with 0.89B high school data on Base1.
- High-school-CPT: CPT with Base1 data and high school data.

Result 4: Both SFT and CPT primarily develop capabilities aligned with their data distributions and SFT's IND learning ability is weaker than that of CPT.

Model	GSM8K	Math	Gaokao	Zhongkao	Average
Base1	28.20	9.48	8.09	30.68	19.11
Middle-school-SFT	22.67 (-5.53)	16.36 (+6.88)	10.21 (+2.12)	**52.28 (+21.60)**	25.38 (+6.27)
Middle-school-CPT	29.42 (+1.22)	15.04 (+5.56)	8.09 (0.00)	**54.71 (+24.03)**	26.81 (+7.70)
High-school-SFT	19.11 (-9.09)	13.48 (+4.00)	**16.60 (+8.51)**	36.78 (+6.10)	21.49 (+2.38)
High-school-CPT	23.96 (-4.24)	13.82 (+4.34)	**22.98 (+14.89)**	34.19 (+3.51)	23.74 (+4.63)

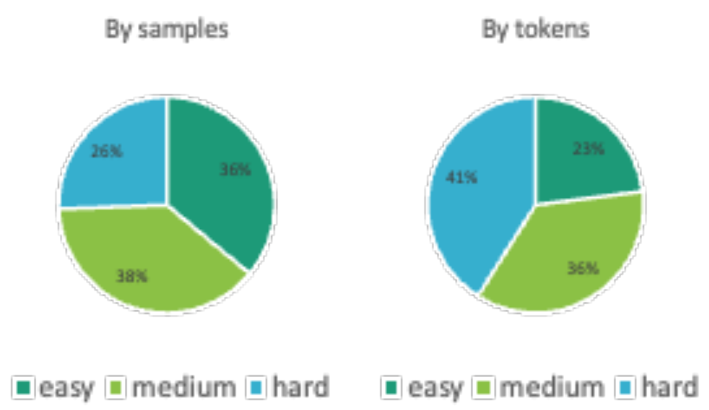
RQ3.3-Data difficulty perspective

(CPT ☺ vs SFT on easy data subset)

- Base1: CPT with 48.3B general corpus and 14.7B math corpus.
- Easy-SFT: SFT with 1.15B easy data subset on Base1.
- Easy-CPT: CPT with Base1 data and easy data subset.

(CPT ☺ vs SFT on hard data subset)

- High-school-SFT: SFT with 0.89B high school data on Base1.
- High-school-CPT: CPT with Base1 data and high school data.



Model	GSM8K	Math	Gaokao	Zhongkao	Average	Easy	Medium	Hard
Base1	28.20	9.48	8.09	30.68	19.11	14.86	6.69	4.85
Easy-SFT	31.31	14.46	14.04	48.30	27.03	22.52 (+7.66)	10.68 (+4.00)	6.94 (+2.09)
Easy-CPT	37.98	15.70	17.02	52.46	30.79	27.61 (+12.75)	13.33 (+6.64)	6.27 (+1.42)
Hard-SFT	31.39	17.40	15.32	54.55	29.66	24.37 (+9.51)	11.93 (+5.24)	6.84 (+1.99)
Hard-CPT	45.79	23.96	26.38	69.89	41.51	35.78 (+20.92)	20.17 (+13.48)	9.32 (+4.47)

Result 5: Providing more challenging problem-solving data enables more effective learning, and this advantage is particularly evident in CPT compared to SFT.

MathGPT-8B

Model	GSM8K	MATH	GAOKAO	ZHONGKAO	Average
General Models					
Llama3-8B	58.38	17.04	13.62	42.61	32.91
Llama3-70B	82.34	38.42	28.09	64.02	53.21
Llama3.1-8B	56.79	19.70	11.49	44.70	33.17
Llama3.1-70B	81.73	39.66	31.06	64.77	54.31
Qwen2-7B	80.44	47.82	27.23	70.45	56.49
Qwen2-72B	86.58	56.88	45.11	73.67	65.56
Qwen2.5-7B	84.61	53.22	45.53	80.30	65.92
Qwen2.5-72B	90.60	59.38	56.60	82.95	72.38
Math-specific Models					
LLEMMA-7B	41.47	18.94	14.89	45.08	30.10
DeepSeekMath-Base-7B	65.73	33.40	23.83	62.69	46.41
Qwen2-Math-7B	80.67	53.02	42.13	77.08	63.22
Qwen2-Math-72B	88.63	61.88	51.91	81.25	70.92
Qwen2.5-Math-7B	85.44	59.10	53.19	78.79	69.13
Qwen2.5-Math-72B	88.70	67.10	53.62	81.63	72.76
(MathGPT-8B)	**81.20**	**60.38**	**60.43**	**80.49**	**70.62**

🌟 CPT on **Llama3-8B** resulted in significant improvement, surpassing version with 70B parameters.

Improvement

✅ MathGPT-8B is trained on only **140B tokens**, while Qwen2.5-Math-7B utilizes **1T tokens** (as reported).

Efficiency

⬆️ Taking challenge problem-solving data in the CPT stage can significantly improve the efficiency of CPT and may be a better training paradigm.

⬇️ Further exploration is needed for the methods used in the post-training phase.