

SAAS: Solving Ability Amplification Strategy for Enhanced Mathematical Reasoning in Large Language Models

Hyeonwoo Kim^{1*}, Gyoungjin Gim^{1*}, Yungi Kim^{1*}, Jihoo Kim¹

Byungju Kim², Wonseok Lee², Chanjun Park^{1†}

{choco_9966, gyoungjin.gim, eddie, jerry, chanjun.park}@upstage.ai

{peyton.kim, jack.lee}@mathpresso.com

At a Glance

- The paper introduces **Solving Ability Amplification Strategy (SAAS)**, a novel approach to enhance **mathematical reasoning** in large language models (LLMs)
- It **combines Chain-of-Thought (CoT) and Program-of-Thought (PoT) learning strategies**, transitioning from logical reasoning to programmatic execution **to amplify problem-solving capabilities**
- The empirical evaluation shows that SAAS outperforms other models on challenging mathematical tasks



Motivation

- The study aims to overcome the limitations of LLMs in mathematical reasoning, which is essential for applications requiring logical thinking and problem-solving
- Current methods like CoT can improve reasoning but lead to computational errors, while PoT focuses on accurate execution but lacks reasoning ability
- The SAAS approach addresses these issues by sequentially integrating CoT and PoT learning



SAAS (Solving Ability Amplification Strategy)

Overview of SAAS

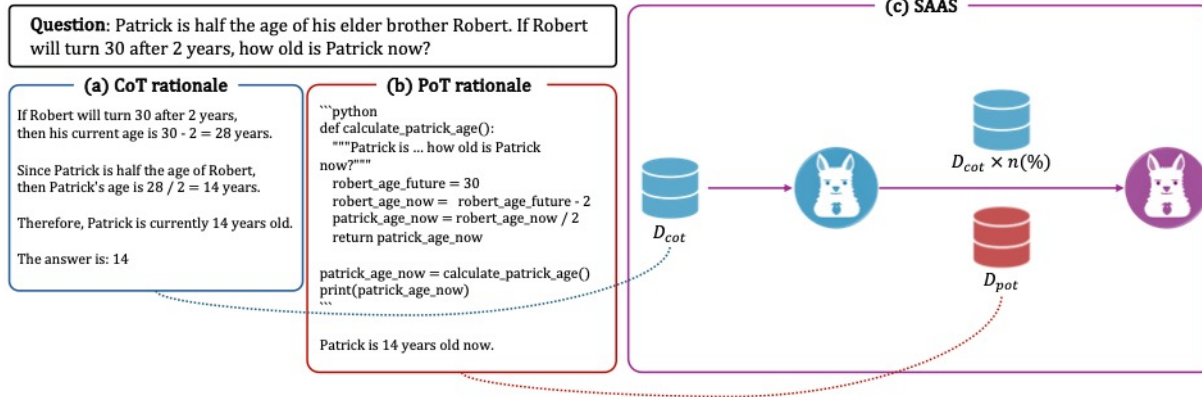


Figure 1: Overview of SAAS (Solving Ability Amplification Strategy) with two core strategies: i) sequential learning strategy; ii) cognitive retention strategy.



Dataset

Training Settings

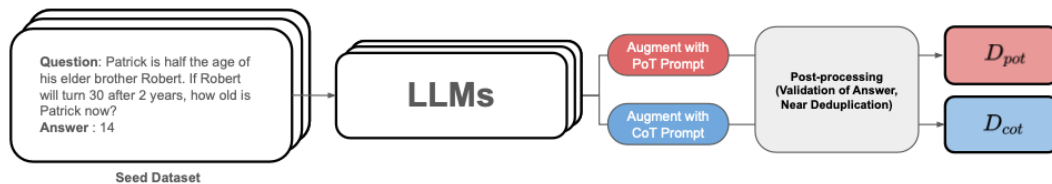


Figure 2: Overall procedure of the synthetic data generation.

Seed Dataset	Rationale	Models	Size
MetaMathQA	CoT	GPT, WizardMath	465K
MATH, GSM8K	CoT	WizardMath	300K
QANDA	CoT	WizardMath	120K
MetaMathQA	PoT	ToRA	60K
MATH, GSM8K	PoT	ToRA	226K
MathInstruct	PoT	ToRA	38K
QANDA	PoT	ToRA	12K

Table 1: Summary of synthetic datasets



Experiments

- **RQ1:** Does SAAS quantitatively outperform its competitors for solving challenging mathematical problems?
- **RQ2:** Are two core strategies of SAAS effective in improving the accuracy?
- **RQ3:** Is SAAS effective in solving not only basic but also challenging mathematical problems?
- **RQ4:** Does sequential learning that transitions from CoT learning to PoT learning help improve both the mathematical reasoning and computational accuracy?



Experiments

RQ1: Does SAAS quantitatively outperform its competitors for solving challenging mathematical problems?

Model	Size	GSM8K	MATH	GSM-Hard	SVAMP	TabMWP	ASDiv	MAWPS	Avg.
General Models									
GPT4	-	92.0	45.2	64.7	93.1	67.1	91.3	97.6	78.3
GPT4 (PAL)	-	94.2	51.8	77.6	94.8	95.9	92.6	97.7	86.4
ChatGPT	-	80.8	35.5	55.9	83.0	69.1	87.3	94.6	72.3
ChatGPT (PAL)	-	78.6	38.7	67.6	77.8	79.9	81.0	89.4	73.3
Claude-2	-	85.2	32.5	-	-	-	-	-	-
PaLM-2	540B	80.7	34.3	-	-	-	-	-	-
LLaMa-2	7B	13.3	4.1	7.8	38.0	31.1	50.7	60.9	29.4
Platypus-2	7B	14.4	5.4	8.6	36.7	26.5	47.9	58.4	28.3
CodE-LaMa (PAL)	7B	34.0	16.6	33.6	59.0	47.3	61.4	79.6	47.4
SOLAR-1	10.7B	25.8	8.0	17.1	59.3	33.6	55.1	68.4	38.1
LLaMa-2	13B	24.3	6.3	13.6	43.1	39.5	56.3	70.4	36.2
Platypus-2	13B	23.7	7.1	14.3	50.7	45.3	55.1	69.6	38.0
CodE-LaMa (PAL)	13B	39.9	19.9	30.0	65.4	59.5	65.3	86.0	53.1
CodE-LaMa (PAL)	34B	53.3	23.9	49.4	71.0	63.1	72.4	91.5	60.7
LLaMa-2	70B	57.8	14.4	36.0	73.6	57.5	76.0	92.4	58.2
Platypus-2	70B	45.9	15.0	24.6	74.3	47.3	72.7	91.1	53.0
Mathematics Domain-Specific Models									
WizardMath	7B	54.9	10.7	20.6	57.3	38.1	59.1	73.7	44.9
MetaMath	7B	66.5	19.8	-	-	-	-	-	-
MuggleMATH	7B	68.4	-	-	-	-	-	-	-
Toolformer	7B	-	-	-	29.4	-	40.4	44.0	-
MathCoder	7B	64.2	23.3	-	-	-	-	-	-
MathCoder-CODE	7B	67.8	30.2	-	-	-	-	-	-
MAmmoTH	7B	53.6	31.5	-	-	-	-	-	-
MAmmoTH-CODE	7B	59.4	33.4	-	-	-	-	-	-
ToRA	7B	68.8	40.1	54.6	68.2	42.4	71.9	88.8	62.4
SAAS	7B	74.3	43.2	58.3	74.3	49.6	72.3	93.6	67.2
ToRA-CODE	7B	72.6	44.6	56.0	70.4	51.6	78.7	91.3	66.5
SAAS-CODE	7B	74.8	45.2	35.1	73.6	44.0	86.4	93.8	70.0
SAAS	10.7B	82.0	50.1	64.9	85.0	72.5	87.5	95.7	76.8
WizardMath	13B	63.9	14.0	28.4	64.3	46.7	65.8	79.7	51.8
MetaMath	13B	72.3	22.4	-	-	-	-	-	-
MuggleMATH	13B	74.0	-	-	-	-	-	-	-
MathCoder	13B	72.6	29.9	-	-	-	-	-	-
MathCoder-CODE	13B	74.1	35.9	-	-	-	-	-	-
MAmmoTH	13B	62.0	34.2	-	-	-	-	-	-
MAmmoTH-CODE	13B	64.7	36.3	-	-	-	-	-	-
ToRA	13B	72.7	43.0	57.3	72.9	47.2	77.2	91.3	65.9
SAAS	13B	76.6	46.2	61.6	77.8	58.2	80.5	94.3	70.7
ToRA-CODE	13B	75.8	48.1	60.5	75.7	65.1	81.4	92.5	71.3
SAAS-CODE	13B	79.4	50.6	61.6	80.6	68.2	84.5	95.4	74.3
MathCoder-CODE	34B	81.7	45.2	-	-	-	-	-	-
MAmmoTH-CODE	34B	72.7	43.6	-	-	-	-	-	-
ToRA-CODE	34B	80.7	50.8	63.7	80.5	70.5	84.2	93.3	74.8
SAAS-CODE	34B	82.9	52.3	64.1	82.8	73.9	83.4	95.2	76.6
SAAS-LEMA	34B	85.4	54.7	67.0	85.2	80.2	87.6	96.6	79.5
WizardMath	70B	81.6	22.7	50.3	80.0	49.8	76.2	86.2	63.8
MetaMath	70B	82.3	26.6	-	-	-	-	-	-
MuggleMATH	70B	82.3	-	-	-	-	-	-	-
MathCoder	70B	83.9	45.1	-	-	-	-	-	-
ToRA	70B	84.3	49.7	67.2	82.7	74.0	86.8	93.8	76.9

Table 2: Accuracies of competitors and our SAAS on the mathematical benchmark datasets. Our SAAS models are shown in purple color.



Experiments

RQ2: Are two core strategies of SAAS effective in improving the accuracy?

Strategy	GSM8K	MATH
Chain-of-Thought (CoT)	69.7	26.9
Program-of-Thought (PoT)	76.8	47.7
Combination of CoT and PoT	<u>79.0</u>	49.2
SAAS	79.4	50.6
without cognitive retention strategy	<u>79.0</u>	<u>49.6</u>
Reverse SAAS	76.8	47.1
without cognitive retention strategy	69.4	27.6

Table 3: Accuracies of different learning strategies. All improvements are statistically significant with p -value ≤ 0.001 .



Experiments

RQ3: Is SAAS effective in solving not only basic but also challenging mathematical problems?

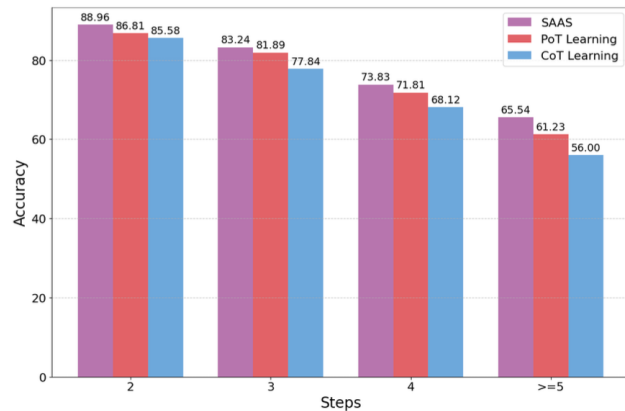


Figure 3: Accuracies on GSM8K with respect to the number of required reasoning steps.



Experiments

RQ4: Does sequential learning that transitions from CoT learning to PoT learning help improve both the mathematical reasoning and computational accuracy?

Question: Benjamin is trying to break a combination lock. This particular type of lock has 5 digits from 0 to 9, and Benjamin just happens to know that there can be no repeated digits in the code for this type of lock. How many valid codes are there?

Answer: 30240

CoT Learning

To find the number of valid codes, we need to consider the number of choices for each digit. For the first digit, there are 9 choices (0-9, excluding 0). For the second digit, there are 8 choices (0-9, excluding the digit already chosen). For the third digit, there are 7 choices (0-9, excluding the digits already chosen). For the fourth digit, there are 6 choices (0-9, excluding the digits already chosen). For the fifth digit, there are 5 choices (0-9, excluding the digits already chosen). Therefore, the total number of valid codes is $9 * 8 * 7 * 6 * 5 = 1260$. The answer is: 1260

PoT Learning

```
python
from sympy import factorial

def valid_codes():
    """Benjamin is trying to break a
    combination lock. This particular type of lock
    has 5 digits from 0 to 9, and Benjamin just
    happens to know that there can be no
    repeated digits in the code for this type of
    lock. How many valid codes are there?"""
    # There are 10 choices for each of the 5
    digits
    num_codes = 10**5

    return num_codes

result = valid_codes()
print(result)

output
100000
The number of valid codes is 100000.
```

SAAS

```
python
from sympy import factorial

def valid_codes():
    """Benjamin is trying to break a
    combination lock. This particular type of lock
    has 5 digits from 0 to 9, and Benjamin just
    happens to know that there can be no
    repeated digits in the code for this type of
    lock. How many valid codes are there?"""
    # There are 10 choices for each of the 5
    digits, and the choices are independent,
    # so we use the product of the number of
    choices for each digit
    valid_codes = factorial(10) / factorial(10 -
    5)

    return valid_codes

result = valid_codes()
print(result)

output
30240
The number of valid codes is 30240.
```

Figure 4: Responses of different learning approaches for a given question-answer pair.



Conclusion

- Prioritizing the learning of mathematical reasoning ability via **Chain-of-Thought (CoT)** learning is helpful for the amplification of problem-solving ability during **Program-of-Thought (PoT) learning**
- For effective sequential learning, it is necessary to employ a **cognitive retention strategy** that incorporates some data samples from the initial phase into the subsequent phase
- **SAAS (Solving Ability Amplification Strategy)**, which progresses from CoT learning to PoT learning with cognitive retention strategy
- Through extensive experiments with the reputable benchmarks, we demonstrated that SAAS consistently and significantly outperforms all competitor, marking a significant advancement in the field of mathematical reasoning in LLMs



Thank You

