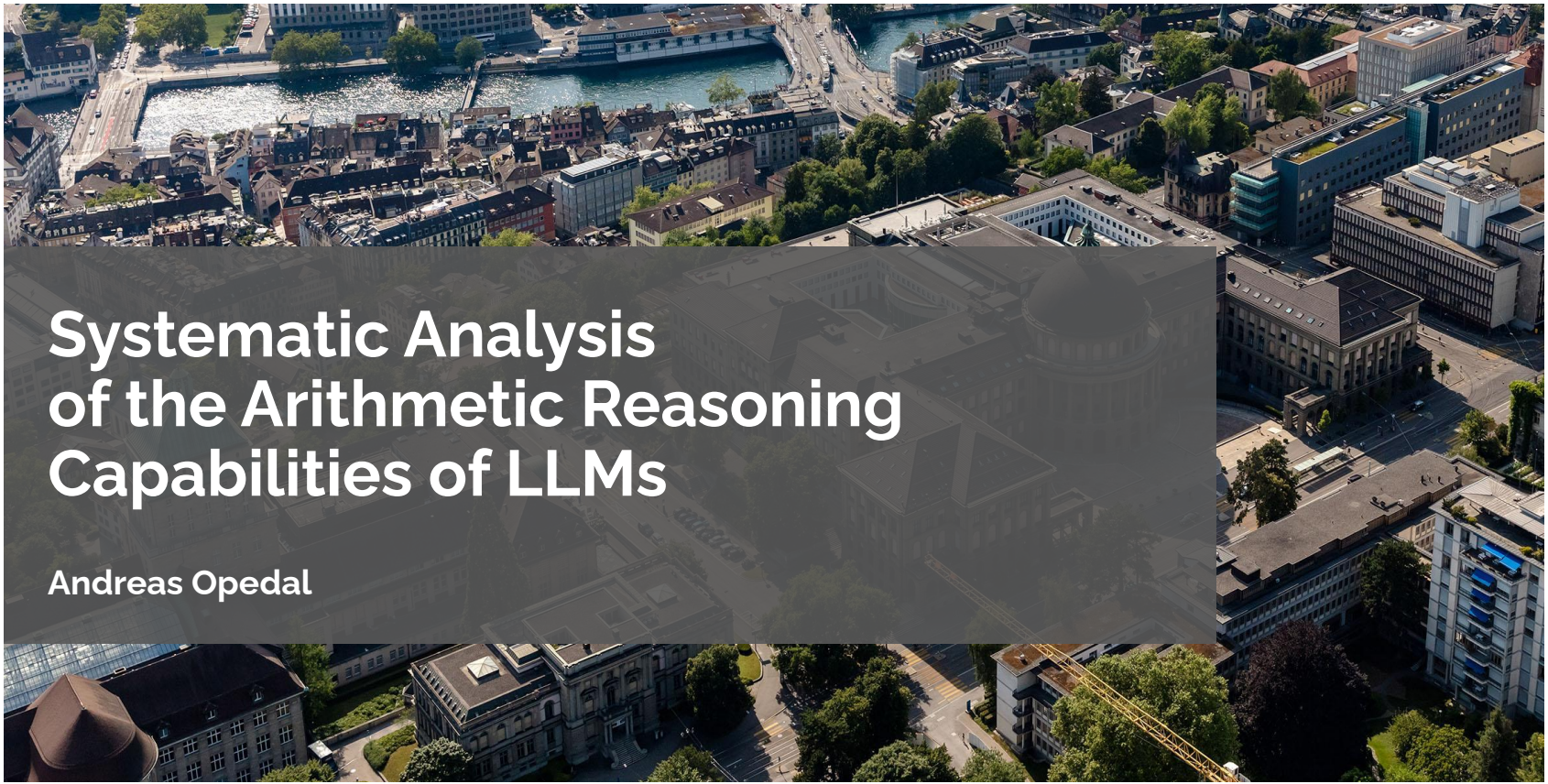


Systematic Analysis of the Arithmetic Reasoning Capabilities of LLMs

Andreas Opedal

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

ETH zürich

Reasoning

- What does it mean to **reason**?

To reason is the capacity of applying logic consciously by drawing conclusions from new or existing information, with the aim of seeking the truth.

– Wikipedia, March 2025

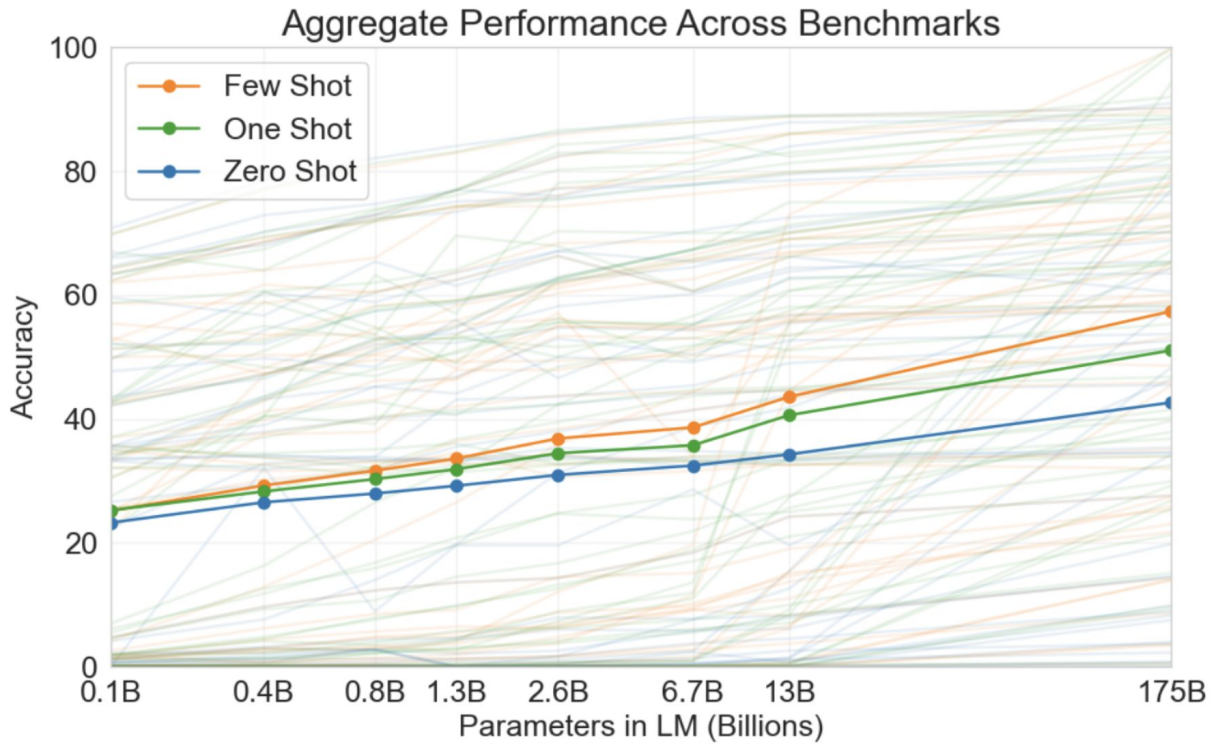
Reasoning

- What does it mean to **reason**?
- Long-standing goal of AI is to build systems that can “reason”
  - Turing’s (1950) test
  - Can machines think? “Thinking” is difficult to define
  - So we replace the question by another: Can a machine perform well in an imitation game?

Reasoning

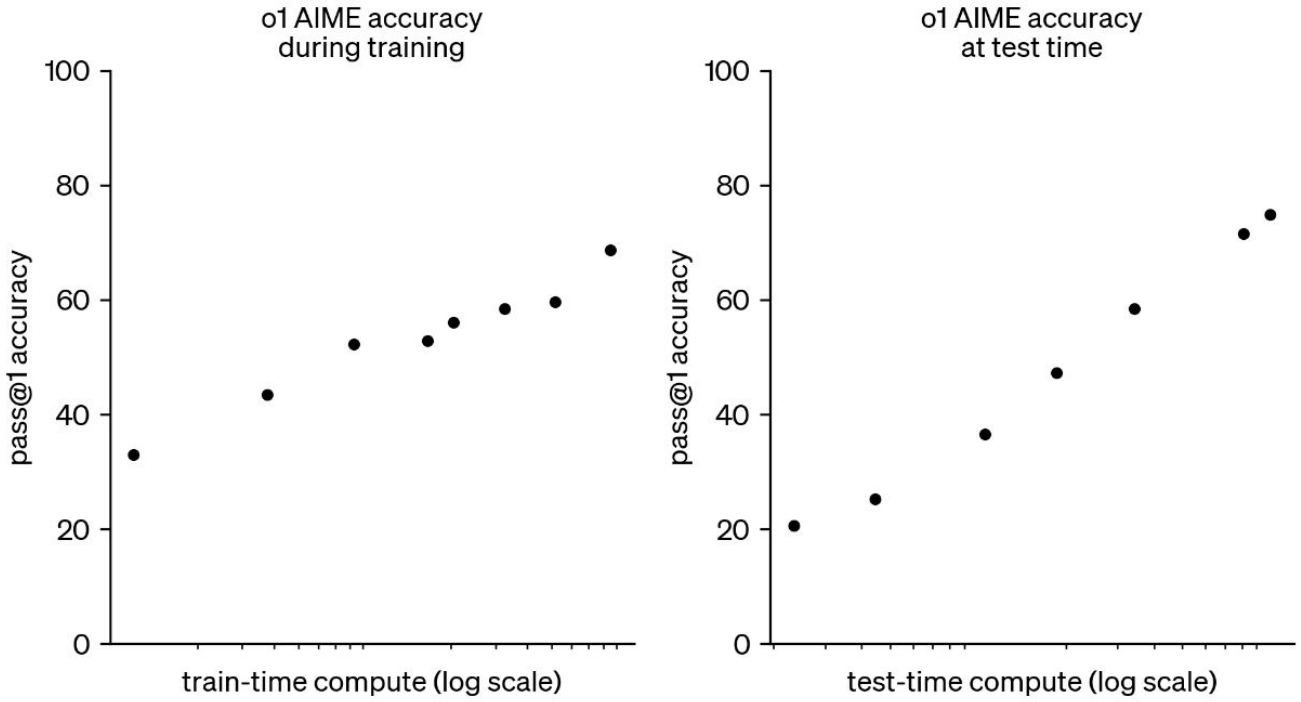
- What does it mean to **reason**?
- Long-standing goal of AI is to build systems that can “reason”
  - Turing’s (1950) test
  - Can machines think? “Thinking” is difficult to define
  - So we replace the question by another: Can a machine perform well in an imitation game?
- Can modern-day ***LLMs*** perform well in an imitation game?

Progress on the Reasoning Imitation Game



(Brown et al., 2020)

Progress on the Reasoning Imitation Game



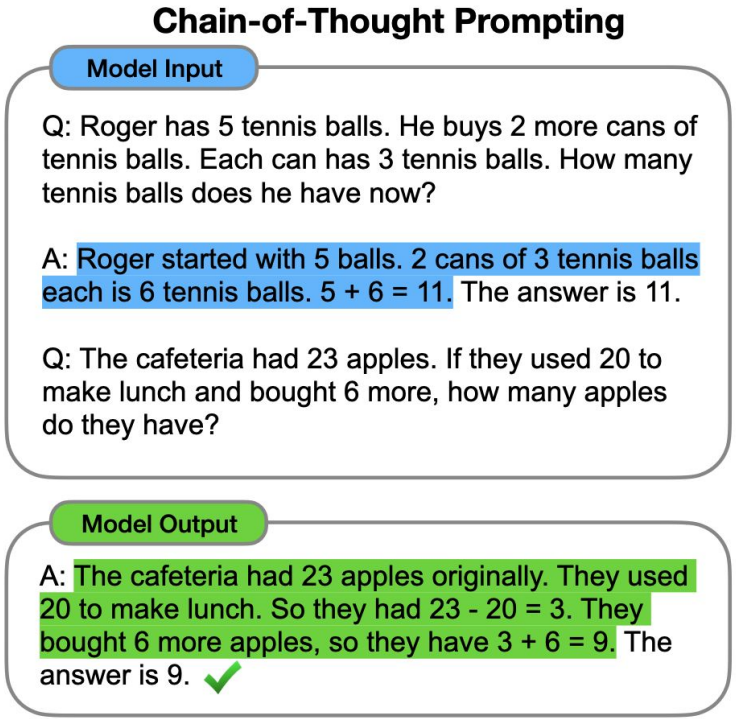
(OpenAI, 2024)

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm



(Wei et al., 2022)

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
    - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

1. What are the characteristics of the problems that the models solve?

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

1. What are the characteristics of the problems that the models solve?
2. Is the dataset truly unseen? Data contamination (Sainz et al., 2023; Deng et al., 2024; *inter alia*)

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

1. What are the characteristics of the problems that the models solve?
2. Is the dataset truly unseen? Data contamination (Sainz et al., 2023; Deng et al., 2024; *inter alia*)
3. Real-world problems may be arbitrarily complex, can the models generalize?

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

We take a (formal) data-centric perspective

World Models for Arithmetic Word Problems

(Opedal et al., 2023)

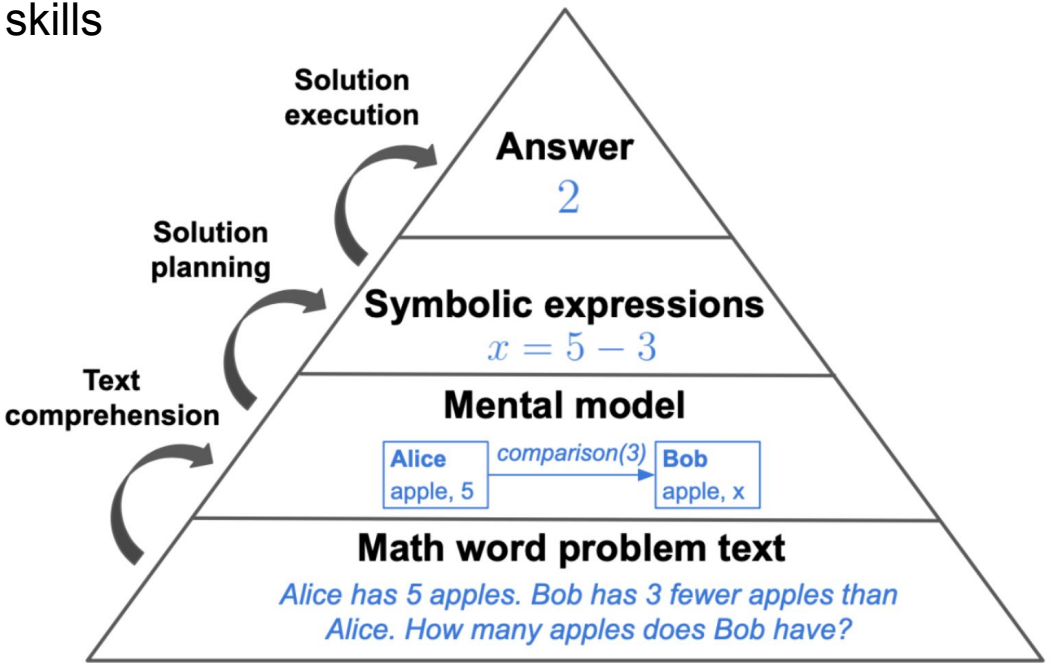
Math Word Problems – What Are They?

- Short narrative text concerning mathematical relationships
- Ends with an interrogative sentence that queries a quantity that can be derived from information in the text

Alice has 5 apples. Bob has 3 fewer apples than Alice. How many apples does Bob have?

Math Word Problems – What Are They?

- Easy (for adults) to understand
- Yet, requires several separate skills



(Nesher and Teubal, 1975; Riley et al., 1983; Kintsch and Greeno, 1985; Hegarty et al., 1995; *inter alia*)

Motivating a Semantic Representation

To understand reasoning capabilities, we want to:

1. Understand the characteristics of the problems
2. Make sure we can generate unseen data

Motivating a Semantic Representation

To understand reasoning capabilities, we want to:

1. Understand the characteristics of the problems
2. Make sure we can generate unseen data

Introduce world-model representation

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

---

6. How many apples does Sam have?
7. Answer: 19

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);
3. container(John, 11, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);
3. container(John, 11, apple);
4. container(Emily, 19, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

---

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);
3. container(John, 11, apple);
4. container(Emily, 19, apple);
5. comp-eq(Lucy, Sam, Emily, John, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1 Isabella has 17 apples.
2 Lucy has 10 more apples than Isabella.
3 John has 11 apples.
4 Emily has 19 apples.
5 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

6 How many apples does Sam have?
7 Answer: 19

World Model

1 container(Isabella, 17, apple);
2 comparison(Lucy, Isabella, 10, apple);
3 container(John, 11, apple);
4 container(Emily, 19, apple);
5 comp-eq(Lucy, Sam, Emily, John, apple);

6 container(Sam, q, apple);

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

---

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);
3. container(John, 11, apple);
4. container(Emily, 19, apple);
5. comp-eq(Lucy, Sam, Emily, John, apple);

---

6. container(Sam, q, apple);
7. container(Sam, 19, apple);

Math Word Problems as Logical Forms

```
transfer(alice, bob, 5, apple)
```

Math Word Problems as Logical Forms

```
transfer(alice, bob, 5, apple)
```

predicate

Relationship expressing arithmetic concept

Math Word Problems as Logical Forms

```
transfer(alice, bob, 5, apple)
```

predicate

Relationship expressing arithmetic concept

properties

Arguments with different meaning

Math Word Problems as Logical Forms

```
transfer(alice, bob, 5, apple)
```

predicate

Relationship expressing arithmetic concept

properties

Arguments with different meaning

Bob gave 5 apples to Alice

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Logical Form		Example Sentences
Predicate	**Properties**	
container	agent=*Alice* quantity=5 entity=*apple* attribute=*red* unit=kg	*Alice has 5 kilograms of red apples.* *Alice owns 5 kilograms of red apples.*
comparison	type=+ agentA=*Alice* agentB=*Bob* quantity=3 entity=*apple*	*Bob has 3 fewer apples than Alice.* *Alice has 3 more apples than Bob.*
transfer	receiver_agent=*Bob* sender_agent=*Alice* quantity=3 entity=*apple*	*Alice gave Bob 3 apples.* *Bob got 3 more apples from Alice.*
rate	agent=*Alice* quantity=4 entityA=*apple* entityB=*basket*	*Each of Alice's baskets holds 4 apples.* *Every basket that Alice has contains 4 apples.*

Math Word Problems as Logical Forms

```
transfer(alice, bob, x, apple)
```

predicate

Relationship expressing arithmetic concept

properties

Arguments with different meaning

How many apples did Bob give to Alice?

Math Word Problems as Logical Forms

- Represent each sentence in the problem as a logical form

Problem Text

1. Isabella has 17 apples.
2. Lucy has 10 more apples than Isabella.
3. John has 11 apples.
4. Emily has 19 apples.
5. The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

---

6. How many apples does Sam have?
7. Answer: 19

World Model

1. container(Isabella, 17, apple);
2. comparison(Lucy, Isabella, 10, apple);
3. container(John, 11, apple);
4. container(Emily, 19, apple);
5. comp-eq(Lucy, Sam, Emily, John, apple);

---

6. container(Sam, q, apple);
7. container(Sam, 19, apple);

Human Biases in Problem Solving

(Opedal*, Stolfo* et al., 2024)

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

1. **What are the characteristics of the problems that the models solve?**
2. **Is the dataset truly unseen? Data contamination** (Sainz et al., 2023; Deng et al., 2024; *inter alia*)
3. Real-world problems may be arbitrarily complex, can the models generalize?

LLMs as cognitive models?

- Simulate responses in human surveys (Argyle et al., 2023)

LLMs as cognitive models?

- Simulate responses in human surveys (Argyle et al., 2023)

- Act as humans in social science experiments (Aher et al., 2023)

LLMs as cognitive models?

- Simulate responses in human surveys (Argyle et al., 2023)
- Act as humans in social science experiments (Aher et al., 2023)
- Be made to model human language acquisition (Warstadt and Bowman, 2022)

LLMs as cognitive models?

- Simulate responses in human surveys (Argyle et al., 2023)
- Act as humans in social science experiments (Aher et al., 2023)
- Be made to model human language acquisition (Warstadt and Bowman, 2022)
- **Simulate human learners** (Macina et al., 2023; Nguyen et al., 2023)

LLMs as cognitive models?

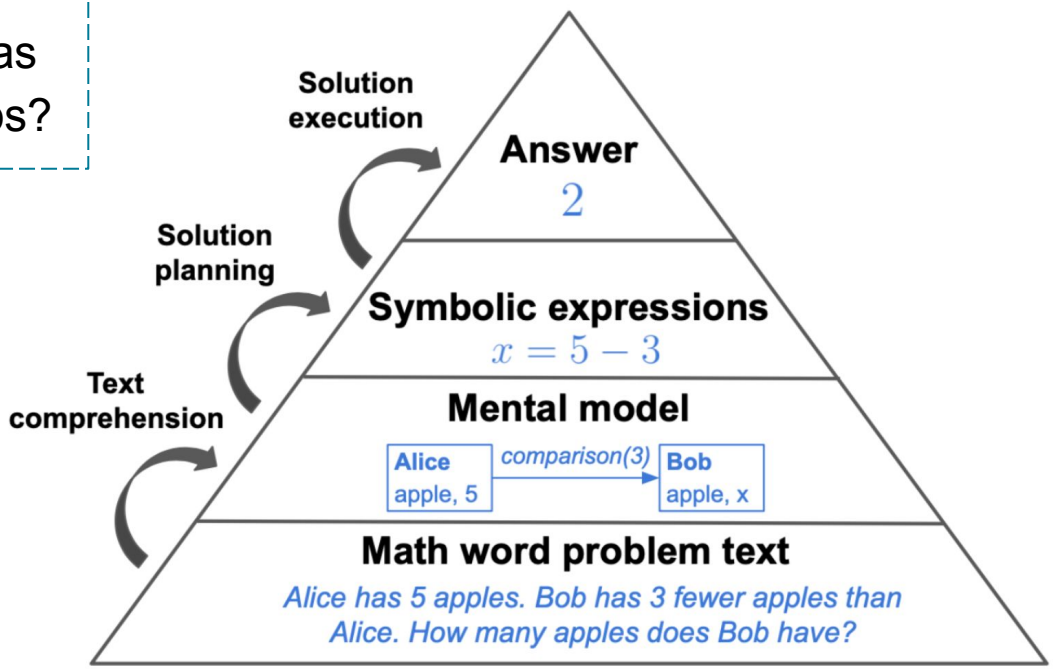
- Simulate responses in human surveys (Argyle et al., 2023)
- Act as humans in social science experiments (Aher et al., 2023)
- Be made to model human language acquisition (Warstadt and Bowman, 2022)
- **Simulate human learners** (Macina et al., 2023; Nguyen et al., 2023)
  - Must remain faithful to human behavior
  - Yet, that is often not the case (Käser and Alexandron, 2023)

The question

Do LLMs exhibit similar biases as human children when solving math word problems?

The question

Do LLMs exhibit similar biases as human children along these steps?



(Nesher and Teubal, 1975; Riley et al., 1983; Kintsch and Greeno, 1985; Hegarty et al., 1995; *inter alia*)

Bias #1: Consistency bias

(Lewis and Mayer, 1987; Stern, 1993)

Text comprehension step

Alice has 5 apples.

How many apples does Bob have?

Bias #1: Consistency bias

(Lewis and Mayer, 1987; Stern, 1993)

Text comprehension step

Alice has 5 apples.

(1) Bob has 3 fewer apples than Alice.

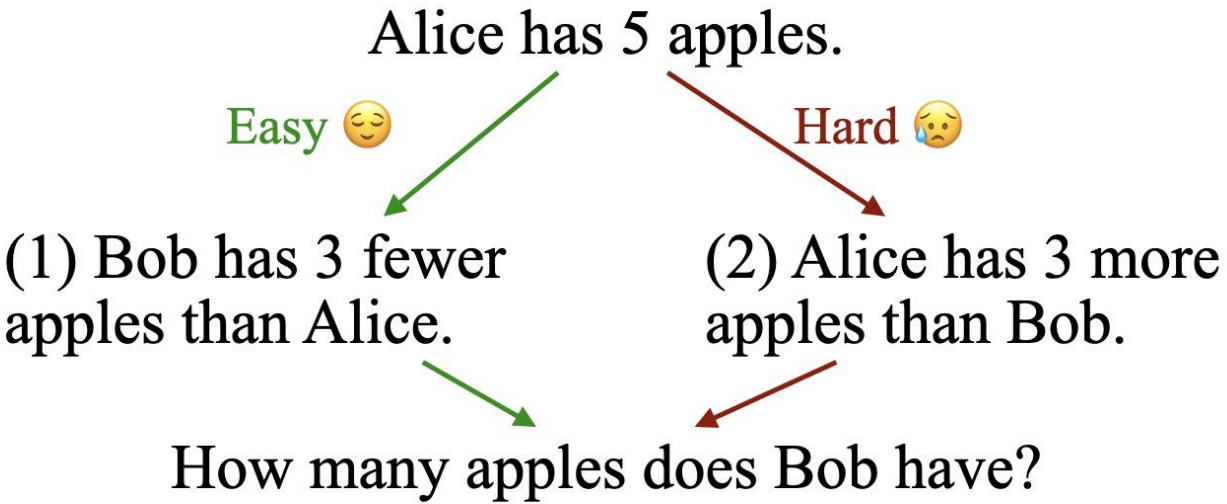
(2) Alice has 3 more apples than Bob.

How many apples does Bob have?

Bias #1: Consistency bias

(Lewis and Mayer, 1987; Stern, 1993)

Text comprehension step



Bias #2: Transfer vs comparison bias

(Riley et al., 1983)

Solution planning step

Alice has 5 apples.

(1) Alice gave 3 apples to Bob.

How many apples does Alice have?

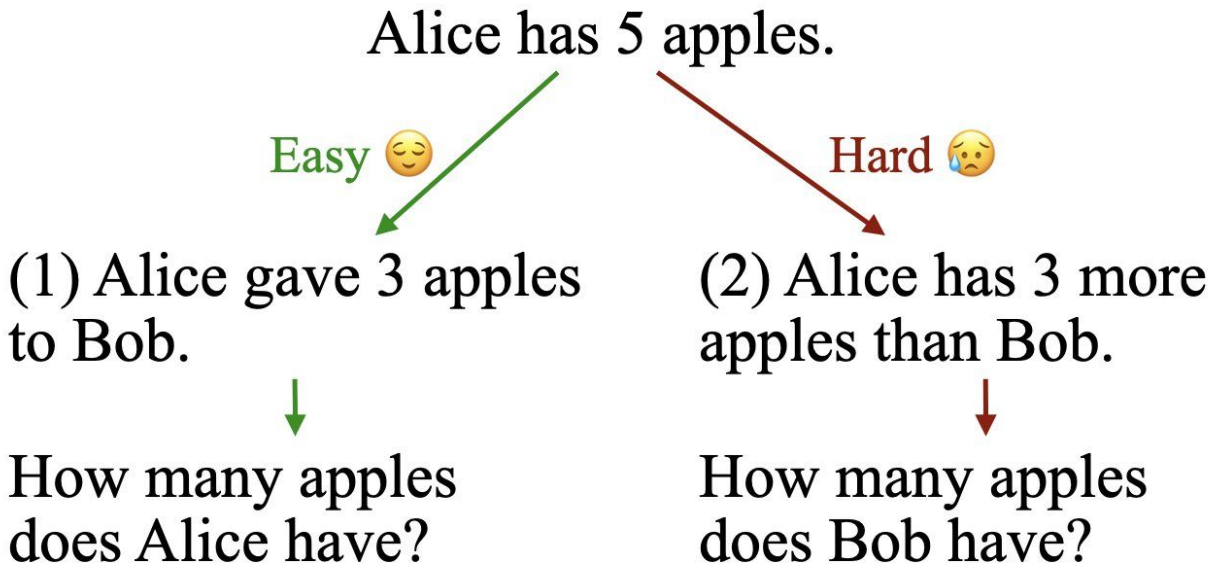
(2) Alice has 3 more apples than Bob.

How many apples does Bob have?

Bias #2: Transfer vs comparison bias

(Riley et al., 1983)

Solution planning step



Bias #3: Carry effect

(Hitch, 1978; Ashcraft et al., 1992)

Solution execution step

$$16 + 7 = 23$$

vs

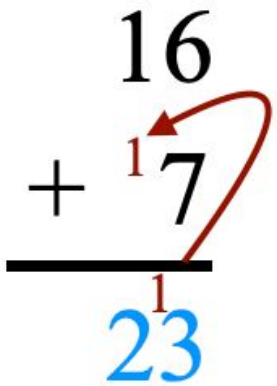
$$16 + 3 = 19$$

Bias #3: Carry effect

(Hitch, 1978; Ashcraft et al., 1992)

Solution execution step

$$16 + 7 = 23$$



Generation method

Problem 1: Test problems from math word problem datasets are likely to have been used in training

Problem 2: We want fine-grained control over the features of the problems, to carry out the tests

Generation method

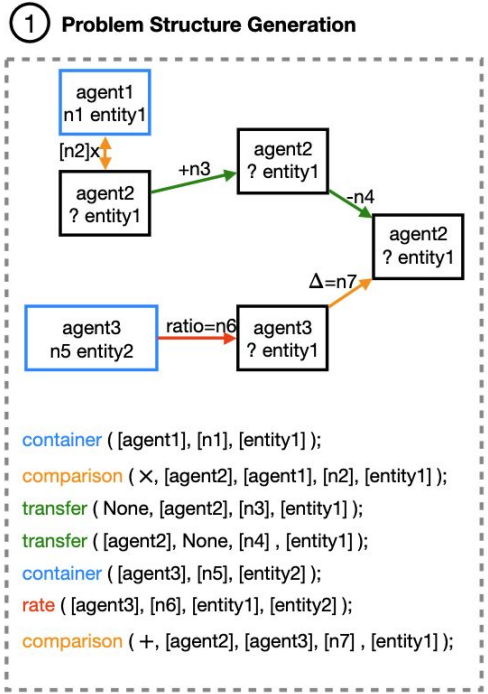
Problem 1: Test problems from math word problem datasets are likely to have been used in training

Problem 2: We want fine-grained control over the features of the problems, to carry out the tests

Solution: Generate our own problems!

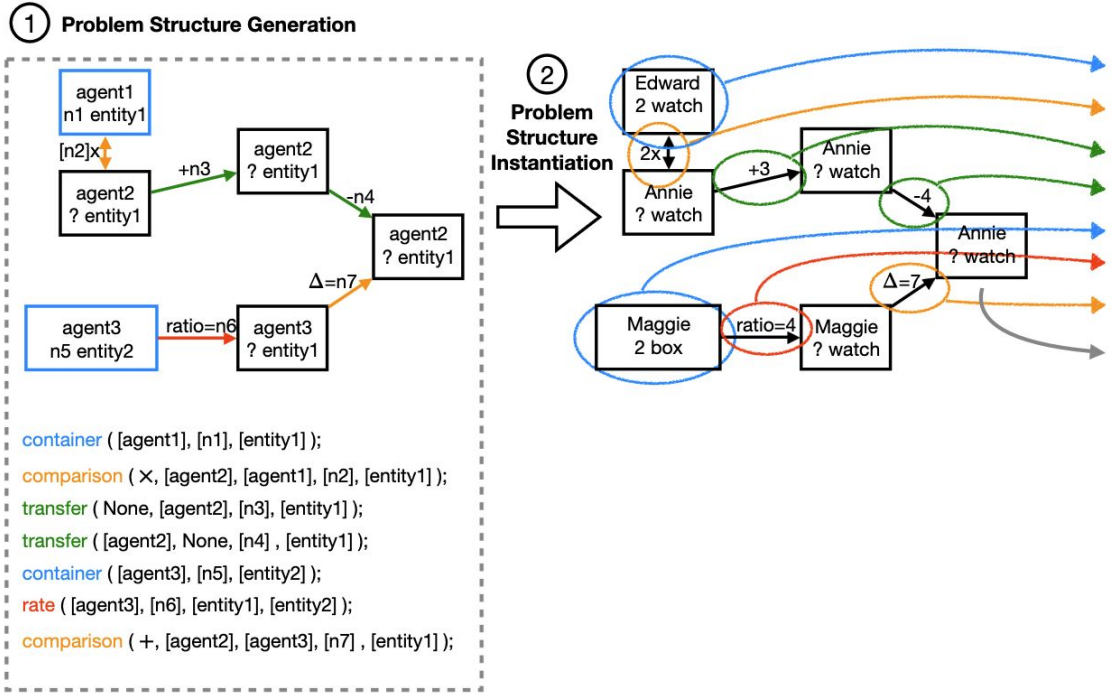
Generation method

Step 1: Problem structure generation



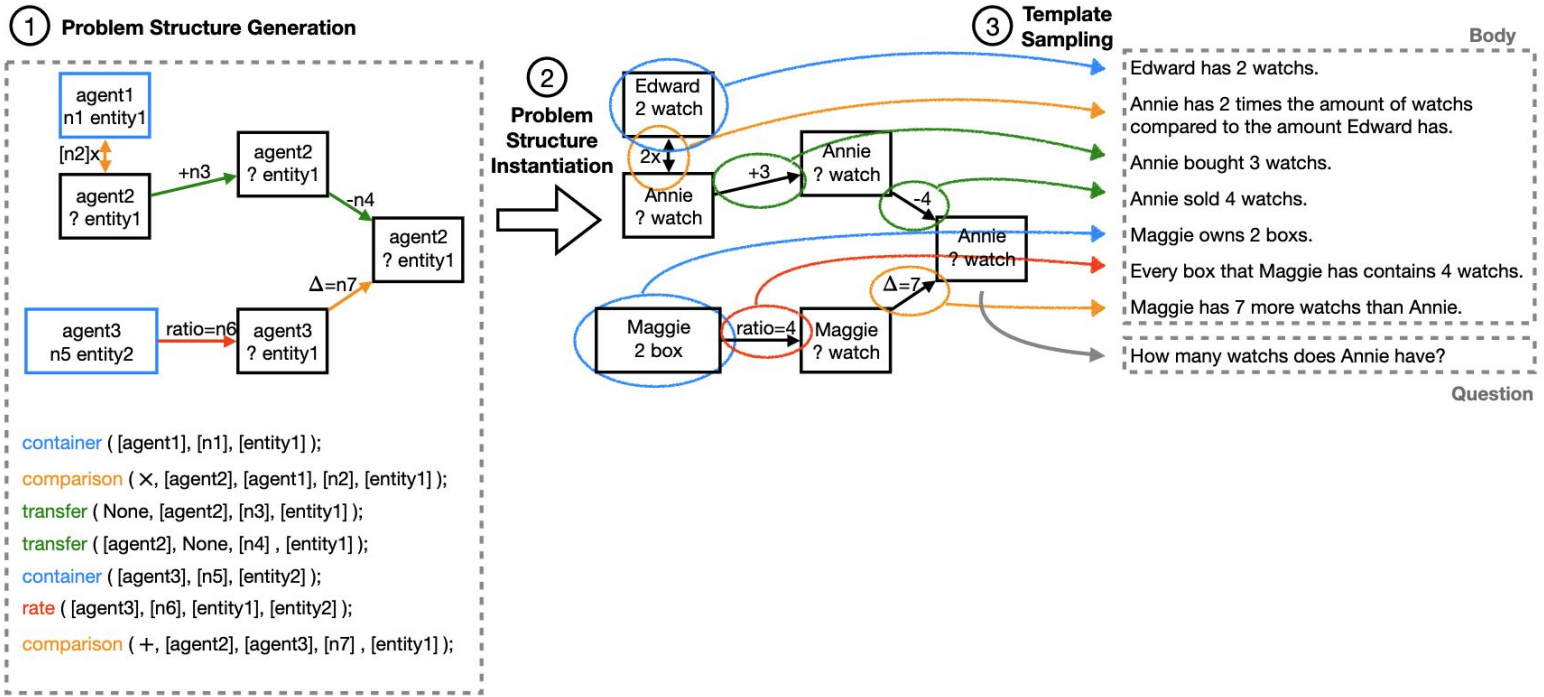
Generation method

Step 2: Problem structure instantiation



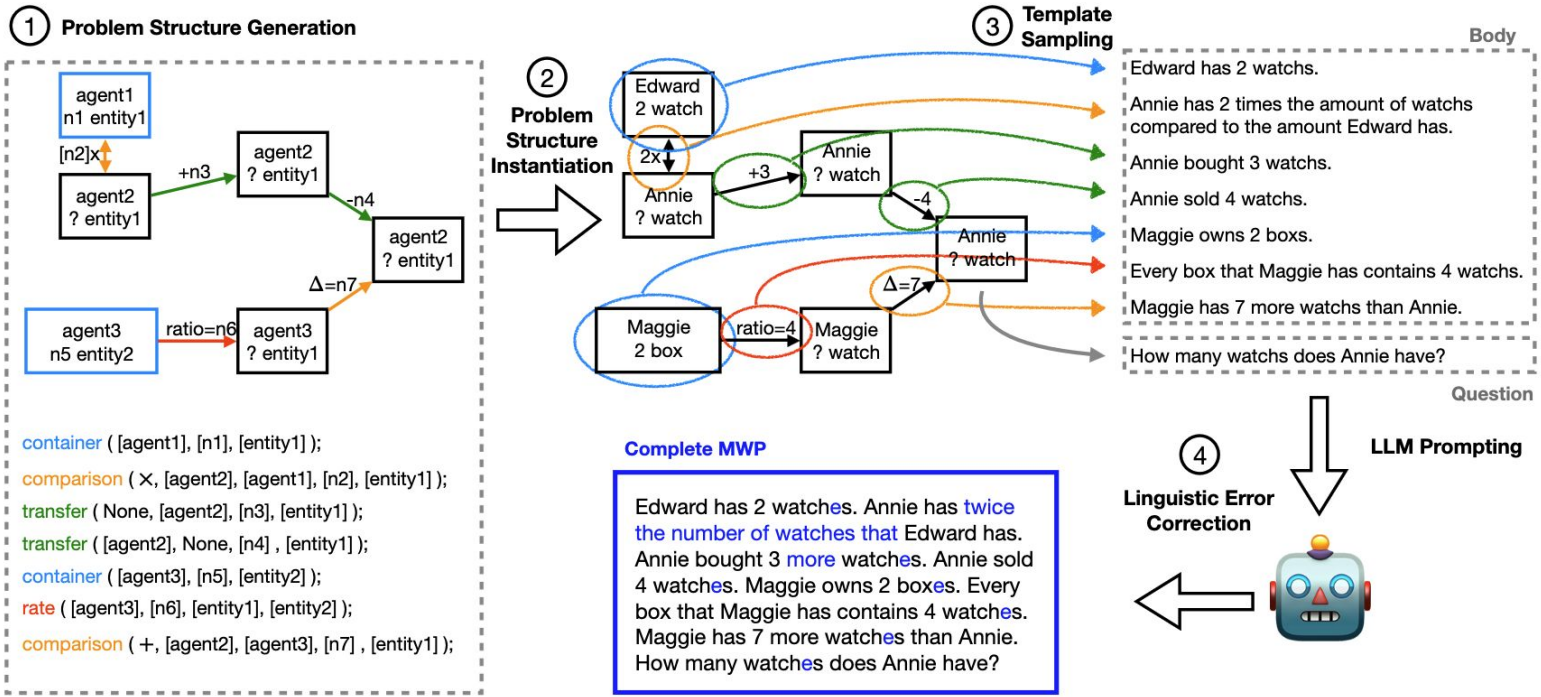
Generation method

Step 3: Template sampling



Generation method

Step 4: Linguistic error correction



Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

- Positive CATEs are consistent with human behavior

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

- Positive CATEs are consistent with human behavior
- Generate a dataset of 500 problem pairs

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

- Positive CATEs are consistent with human behavior
- Generate a dataset of 500 problem pairs
- Zero-shot inference, greedy decoding

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

- Positive CATEs are consistent with human behavior
- Generate a dataset of 500 problem pairs
- Zero-shot inference, greedy decoding
- Direct prompting and chain-of-thought prompting

Experimental setup

- Want a causal effect of a problem feature **X** on LLM performance **Y**
- Generate problems in pairs, **X=x** and **X=x'**, and estimate CATE:

$$\mathbb{E}[Y(x) - Y(x') \mid Z]$$

- Positive CATEs are consistent with human behavior
- Generate a dataset of 500 problem pairs
- Zero-shot inference, greedy decoding
- Direct prompting and chain-of-thought prompting
- Pretrained-only and instruction-tuned models: Llama2 7B/13B, Mistral 7B, Mixtral 8x7B, GPT-3.5 Turbo, GPT-4 Turbo

Experiments: Consistency bias

- Problem specification:

$$\texttt{container} \circ \underbrace{(\texttt{transfer}|\texttt{rate}) \circ \cdots \circ (\texttt{transfer}|\texttt{rate})}_{0-2 \text{ times}} \circ$$

$$\texttt{comparison} \circ \underbrace{(\texttt{transfer}|\texttt{rate}) \circ \cdots \circ (\texttt{transfer}|\texttt{rate})}_{0-2 \text{ times}};$$

- Only comparison sentence varies between the two problems
- Addition, subtraction, multiplication, division

Results: Consistency bias

Mode	Model	Consistency bias (§5.2)			
		Accuracy (%)			p-value
		Co	InCo	CATE	
Direct	LLaMA2 7B	9.6	4.8	**4.8**	<0.001
	LLaMA2 13B	17.2	14.0	**3.2**	0.006
	LLaMA2 70B	24.0	16.2	**7.8**	<0.001
	Mistral 7B	17.8	12.0	**5.8**	<0.001
	Mixtral 8x7B	23.0	17.0	**6.0**	<0.001
	LLaMA2 7B Chat	14.2	10.8	**3.4**	0.009
	LLaMA2 13B Chat	16.4	11.8	**4.6**	<0.001
	LLaMA2 70B Chat	16.4	14.8	1.6	0.158
	Mistral 7B Instr.	17.6	14.2	**3.4**	0.008
	Mixtral 8x7B Instr.	23.4	21.8	1.6	0.195
	GPT-3.5 Turbo	32.2	22.8	**9.4**	<0.001
CoT	LLaMA2 7B	16.4	6.0	**10.4**	<0.001
	LLaMA2 13B	30.2	8.6	**21.6**	<0.001
	LLaMA2 70B	40.2	24.0	**16.2**	<0.001
	Mistral 7B	36.4	16.8	**19.6**	<0.001
	Mixtral 8x7B	62.4	42.2	**20.2**	<0.001
	LLaMA2 7B Chat	66.8	38.6	**28.2**	<0.001
	LLaMA2 13B Chat	67.0	28.6	**38.4**	<0.001
	LLaMA2 70B Chat	82.8	61.4	**21.4**	<0.001
	Mistral 7B Instr.	61.8	33.6	**28.2**	<0.001
	Mixtral 8x7B Instr.	85.4	71.6	**13.8**	<0.001
	GPT-3.5 Turbo	89.2	87.8	1.4	0.380
	GPT-4 Turbo	90.4	72.4	**18.0**	<0.001

Experiments: Transfer vs comparison bias

- Problem specification(s):

$$\texttt{container} \circ \underbrace{\texttt{transfer} \circ \cdots \circ \texttt{transfer}}_{1-5 \text{ times}};$$

$$\texttt{container} \circ \underbrace{\texttt{comparison} \circ \cdots \circ \texttt{comparison}}_{1-5 \text{ times}};$$

- Same symbolic expressions, same named entities

Results: Transfer vs comparison bias

Mode	Model	Transfer vs comparison bias (§5.3)			
		Accuracy (%)			p-value
		T	C	CATE	
Direct	LLaMA2 7B	21.8	13.0	**8.8**	<0.001
	LLaMA2 13B	28.6	20.0	**8.6**	<0.001
	LLaMA2 70B	45.4	26.8	**18.6**	<0.001
	Mistral 7B	34.0	20.4	**13.6**	<0.001
	Mixtral 8x7B	42.2	30.4	**11.8**	<0.001
	LLaMA2 7B Chat	20.2	15.8	**4.4**	0.005
	LLaMA2 13B Chat	25.4	18.2	**7.2**	<0.001
	LLaMA2 70B Chat	32.4	20.0	**12.4**	<0.001
	Mistral 7B Instr.	28.0	21.8	**6.2**	<0.001
	Mixtral 8x7B Instr.	42.6	28.0	**14.6**	<0.001
	GPT-3.5 Turbo	61.0	33.4	**27.6**	<0.001
CoT	LLaMA2 7B	18.8	13.6	**5.2**	0.009
	LLaMA2 13B	37.8	13.2	**24.6**	<0.001
	LLaMA2 70B	63.8	33.0	**30.8**	<0.001
	Mistral 7B	49.8	58.8	**-9.0**	0.004
	Mixtral 8x7B	68.6	65.0	3.6	0.206
	LLaMA2 7B Chat	69.6	40.8	**28.8**	<0.001
	LLaMA2 13B Chat	79.4	48.0	**31.4**	<0.001
	LLaMA2 70B Chat	99.0	76.2	**22.8**	<0.001
	Mistral 7B Instr.	83.4	52.0	**31.4**	<0.001
	Mixtral 8x7B Instr.	98.2	83.8	**14.4**	<0.001
	GPT-3.5 Turbo	97.0	93.0	**4.0**	0.003
	GPT-4 Turbo	99.2	91.4	**7.8**	<0.001

Experiments: Carry effect

- One-step additive comparison problems:

container ∘ comparison;

- Operands and answer are all three-digit numbers (like Fürst and Hitch, 2000)
- One problem has no carry, other has at least one (unit and/or tens)

Results: Carry effect

Mode	Model	Carry effect (§5.4)			
		Accuracy (%)			*p*-value
		NCa	**Ca**	CATE	
Direct	LLaMA2 7B	64.8	60.0	**4.8**	0.009
	LLaMA2 13B	72.2	67.2	5.0	0.030
	LLaMA2 70B	95.2	96.2	1.0	0.380
	Mistral 7B	72.4	72.0	0.4	0.835
	Mixtral 8x7B	95.4	93.6	1.8	0.117
	LLaMA2 7B Chat	61.2	54.2	7.0	0.012
	LLaMA2 13B Chat	65.6	59.6	6.0	0.018
	LLaMA2 70B Chat	96.4	97.0	-0.6	0.578
	Mistral 7B Instr.	78.0	78.6	-0.6	0.802
	Mixtral 8x7B Instr.	95.8	96.4	-0.6	0.578
	GPT-3.5 Turbo	99.6	99.4	0.2	0.320
CoT	LLaMA2 7B	33.2	38.8	-5.6	0.006
	LLaMA2 13B	33.8	33.4	0.4	0.833
	LLaMA2 70B	68.6	67.6	1.0	0.850
	Mistral 7B	73.2	71.0	2.2	0.283
	Mixtral 8x7B	79.8	79.8	0.0	1.000
	LLaMA2 7B Chat	72.4	71.0	1.4	0.514
	LLaMA2 13B Chat	73.8	78.6	-4.8	0.017
	LLaMA2 70B Chat	97.0	95.8	1.2	0.180
	Mistral 7B Instr.	78.6	75.6	3.0	0.162
	Mixtral 8x7B Instr.	97.0	94.6	2.4	0.014
	GPT-3.5 Turbo	97.8	98.2	-0.4	0.580
	GPT-4 Turbo	99.6	99.6	0.0	-

Summary

- Biases in text comprehension and solution planning, but not solution execution

Summary

- Biases in text comprehension and solution planning, but not solution execution
- Why?
  - Training data influenced by adult thinking
  - Perhaps the carry effect is less prevalent in adults

Summary

- Biases in text comprehension and solution planning, but not solution execution
- Why?
  - Training data influenced by adult thinking
  - Perhaps the carry effect is less prevalent in adults
- Chain of thought amplifies biases in most settings

Summary

- Biases in text comprehension and solution planning, but not solution execution
- Why?
  - Training data influenced by adult thinking
  - Perhaps the carry effect is less prevalent in adults
- Chain of thought amplifies biases in most settings
- Implication: Student model practitioners should exercise care

A Proof System for Arithmetic Word Problems

(Opedal*, Shirakami* et al., 2025)

Progress on the Reasoning Imitation Game

- Standard evaluation paradigm
  - Compare models in terms of answer accuracy on benchmark datasets

However, our understanding is restricted…

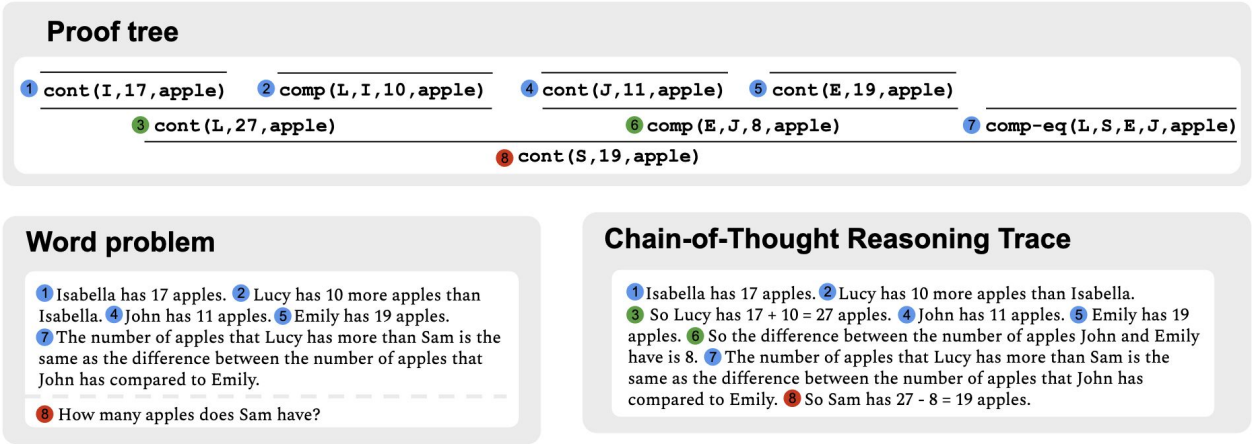
1. **What are the characteristics of the problems that the models solve?**
2. **Is the dataset truly unseen? Data contamination** (Sainz et al., 2023; Deng et al., 2024; *inter alia*)
3. **Real-world problems may be arbitrarily complex, can the models generalize?**

MathGAP

- Framework for evaluating Mathematical Generation on Arithmetic Proofs

MathGAP

- Framework for evaluating Mathematical Generation on Arithmetic Proofs
- Idea: Generate problems by sampling proof trees



Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree

Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree
- Inference rules govern what proof steps are sound in arithmetic reasoning

$$\frac{L_1 \qquad L_2 \qquad \cdots \qquad L_N}{L}$$

Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree
- Say we know:
  - Isabella has 17 apples
  - Lucy has 10 more apples than Isabella

```
cont(Isabella, 17, apple)
```

```
comp(Lucy, Isabella, 10, apple)
```

Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree
- Say we know:
  - Isabella has 17 apples

    `cont(Isabella, 17, apple)`
  - Lucy has 10 more apples than Isabella

    `comp(Lucy, Isabella, 10, apple)`
- Then we can infer:
  - Lucy has 27 apples

    `cont(Lucy, 17 + 10, apple)`

Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree
- Say we know:
  - Isabella has 17 apples

$$\texttt{cont(Isabella, 17, apple)}$$

  - Lucy has 10 more apples than Isabella

$$\texttt{comp(Lucy, Isabella, 10, apple)}$$

- Then we can infer:

$$\frac{\texttt{cont(Isabella, 17, apple)} \quad \texttt{comp(Lucy, Isabella, 10, apple)}}{\texttt{cont(Lucy, 17 + 10, apple)}}$$

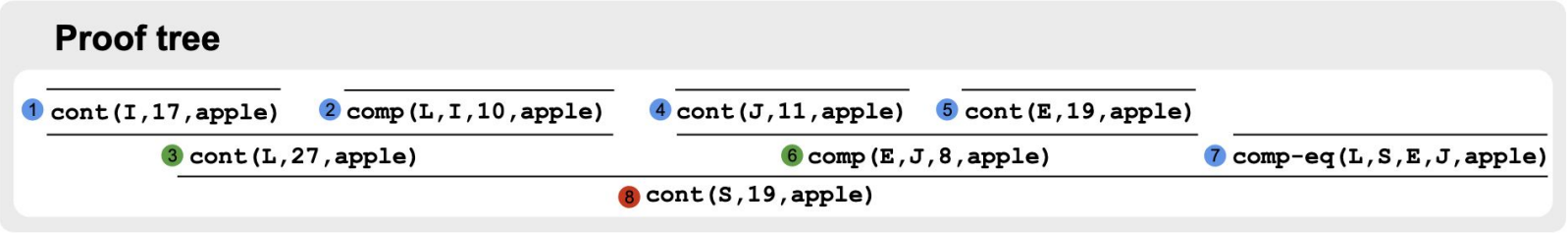
Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree

Inference Rules	Example Sentences
$\frac{\texttt{cont(a, } q_1\texttt{, e)} \quad \texttt{comp(b, a, } q_2\texttt{, e)}}{\texttt{cont(b, } q_1 + q_2\texttt{, e)}}$	*Alice has 3 apples. Bob has 2 more apples than Alice.* $\vdash$ *Bob has 5 apples.*
$\frac{\texttt{cont(a, } q_1\texttt{, e)} \quad \texttt{transfer(a, b, } q_2\texttt{, e)}}{\texttt{cont(a, } q_1 + q_2\texttt{, e)}}$	*Alice has 3 apples. Bob gave 2 apples to Alice.* $\vdash$ *Alice has 5 apples.*
$\frac{\texttt{cont(a, } q_1\texttt{, e)} \quad \texttt{cont(b, } q_2\texttt{, e)}}{\texttt{comp(b, a, } q_2 - q_1\texttt{, e)}}$	*Alice has 3 apples. Bob has 5 apples.* $\vdash$ *Bob has 2 more apples than Alice.*
$\frac{\texttt{cont(a}_1\texttt{, } q_1\texttt{, e)} \ldots \texttt{cont(a}_n\texttt{, } q_n\texttt{, e) partwhole(}\wedge_{i=1}^{n}\texttt{a}_i\texttt{, a}_1\texttt{, ..., a}_n\texttt{, f, e)}}{\texttt{cont(}\wedge_{i=1}^{n}\texttt{a}_i\texttt{, } \sum_{i=1}^{n} q_i\texttt{, f)}}$	*Alice has 3 apples. Bob has 5 apples. Alice and Bob combine their fruits.* $\vdash$ *Alice and Bob have 8 fruits.*
$\frac{\texttt{cont(a, } q_1\texttt{, e)} \quad \texttt{comp(d, c, } q_2\texttt{, e)} \quad \texttt{comp-eq(b, a, d, c)}}{\texttt{cont(b, } q_1 + q_2\texttt{, e)}}$	*Alice has 7 apples. David has 2 more apples than Charlie. The number of apples that Bob has more than Alice is the same as the difference between the number of apples that David and Charlie have.* $\vdash$ *Bob has 9 apples.*

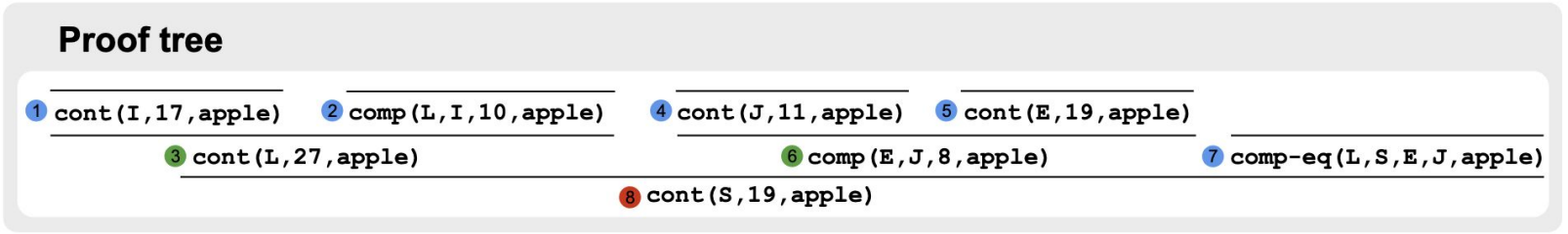
Chain-of-Thought Solutions as Proof Trees

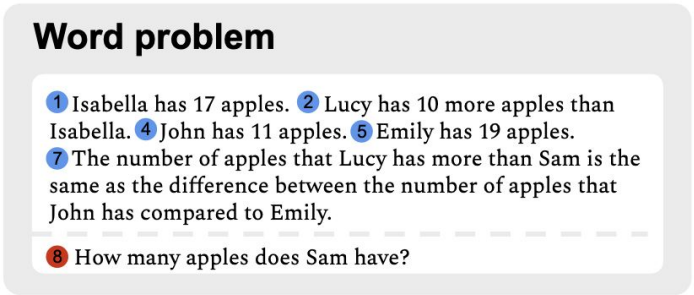
- Use the logical forms as node labels in a proof tree



Chain-of-Thought Solutions as Proof Trees

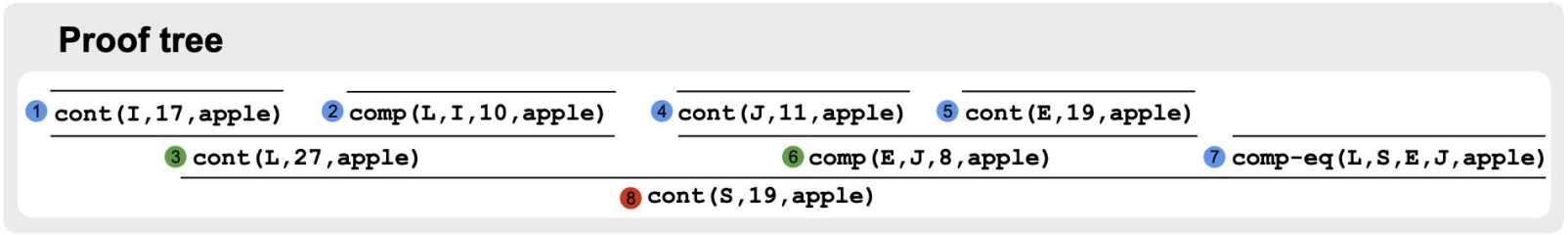
- Use the logical forms as node labels in a proof tree





Chain-of-Thought Solutions as Proof Trees

- Use the logical forms as node labels in a proof tree



Word problem

1 Isabella has 17 apples. 2 Lucy has 10 more apples than Isabella. 4 John has 11 apples. 5 Emily has 19 apples. 7 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

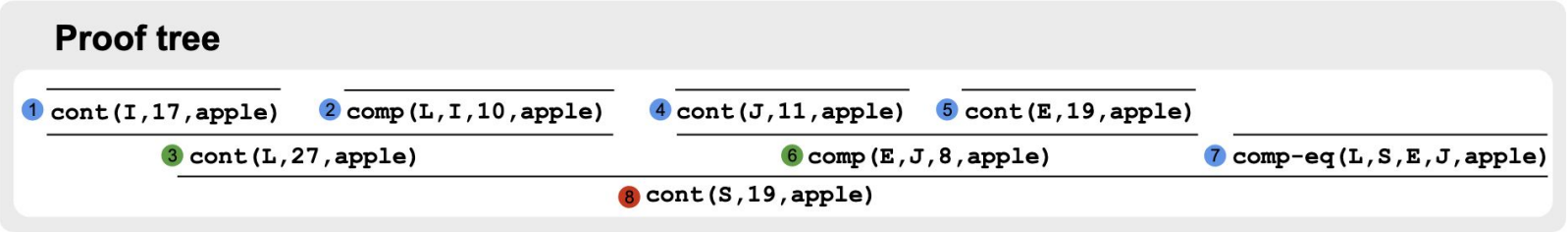
8 How many apples does Sam have?

Chain-of-Thought Reasoning Trace

1 Isabella has 17 apples. 2 Lucy has 10 more apples than Isabella. 3 So Lucy has 17 + 10 = 27 apples. 4 John has 11 apples. 5 Emily has 19 apples. 6 So the difference between the number of apples John and Emily have is 8. 7 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily. 8 So Sam has 27 - 8 = 19 apples.

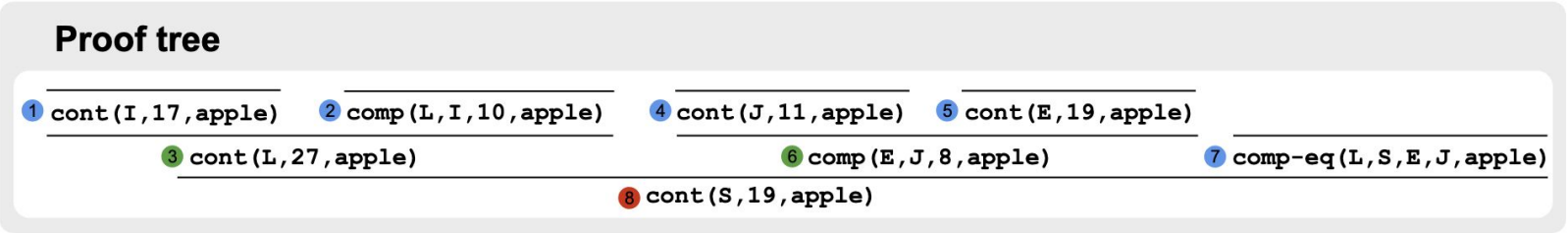
Chain-of-Thought Solutions as Proof Trees

- Can characterize complexity of reasoning in terms of:



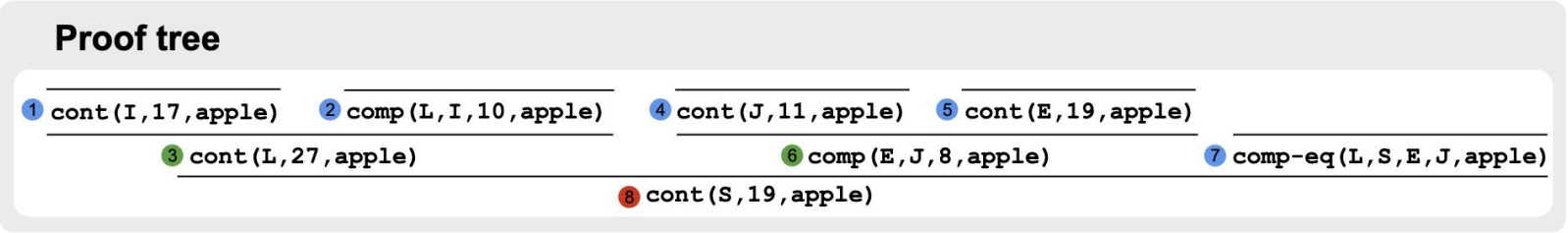
Chain-of-Thought Solutions as Proof Trees

- Can characterize complexity of reasoning in terms of:
    - Depth of the tree: *how many nodes between axioms and answer*



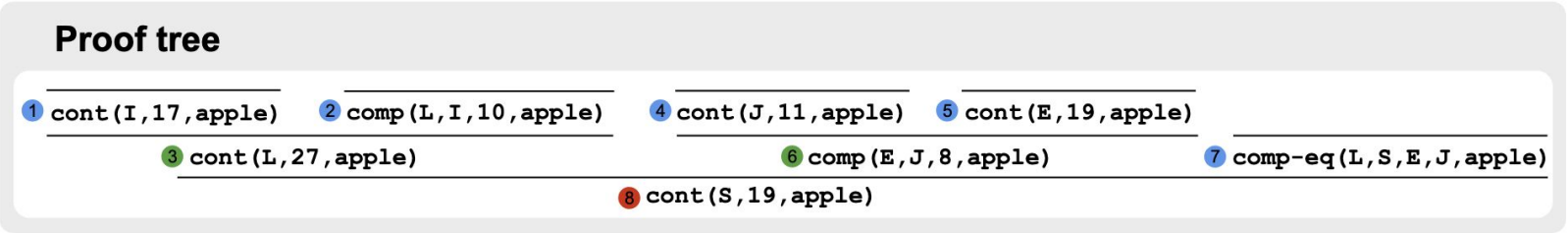
Chain-of-Thought Solutions as Proof Trees

- Can characterize complexity of reasoning in terms of:
  - Depth of the tree
  - Width of the tree: *how many axioms given in the problem*



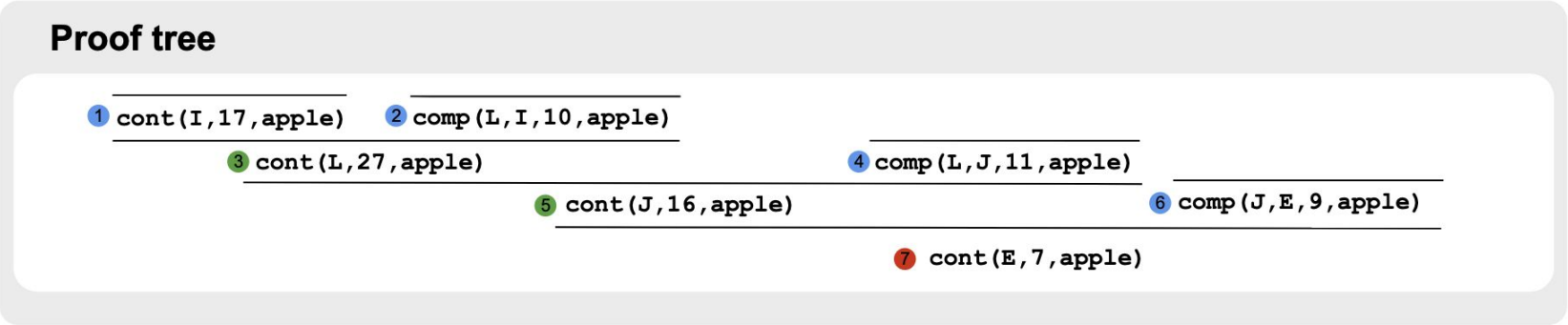
Chain-of-Thought Solutions as Proof Trees

- Can characterize complexity of reasoning in terms of:
  - Depth of the tree
  - Width of the tree
  - Shape of the tree: *how are the axioms combined to get to the answer*



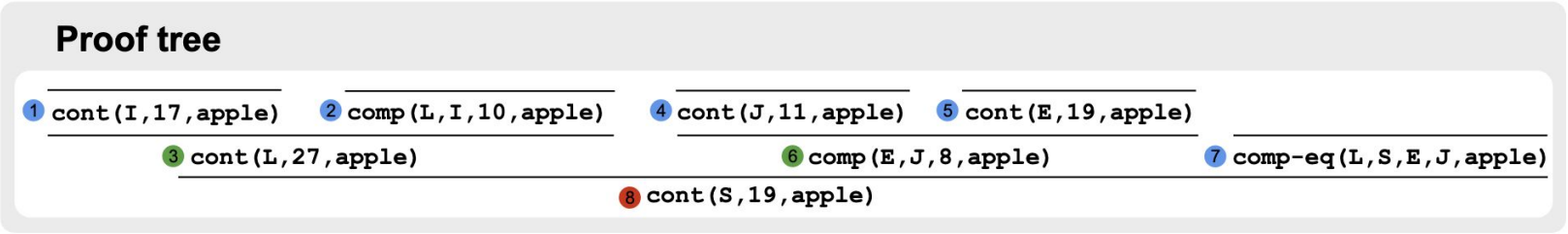
Chain-of-Thought Solutions as Proof Trees

- Shape of the tree:
  - **Linear:** every proof step takes at most one premise that is not an axiom



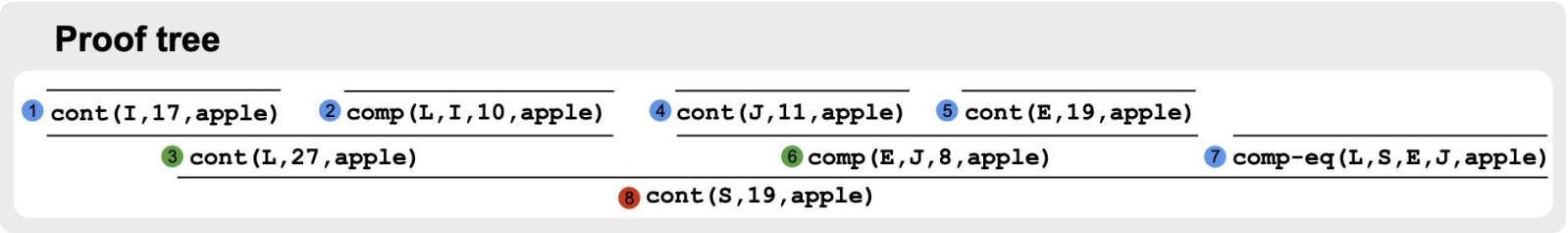
Chain-of-Thought Solutions as Proof Trees

- Shape of the tree:
  - Linear
  - **Nonlinear**



Chain-of-Thought Solutions as Proof Trees

- Can characterize complexity of reasoning in terms of:
  - Depth of the tree
  - Width of the tree
  - Shape of the tree (linear and nonlinear)
  - Ordering of the leaf nodes: *in which order are the axioms presented*



Generating Problems

Step 1: Given a root logical form, sample a ***proof tree*** by iteratively applying inference rules until a stopping criterion has been reached.

Problem specification:
- Nonlinear
- Depth: 2
- Width: 5
- Canonical ordering

Available logical form templates:

```
1. cont([agent],[quantity],[entity])
2. comp([agent1],[agent2],[quantity],[entity])
3. comp-eq([agent1],[agent2],[agent3],[agent4],[entity])
[...]
```

Available inference rules:

```
1. cont(...) comp(...) ⊢ cont(...)
2. cont(...) cont(...) ⊢ comp(...)
3. cont(...) comp(...) comp-eq(...) ⊢ cont(...)
[...]
```

Generating Problems

Step 1: Given a root logical form, sample a ***proof tree*** by iteratively applying inference rules until a stopping criterion has been reached.

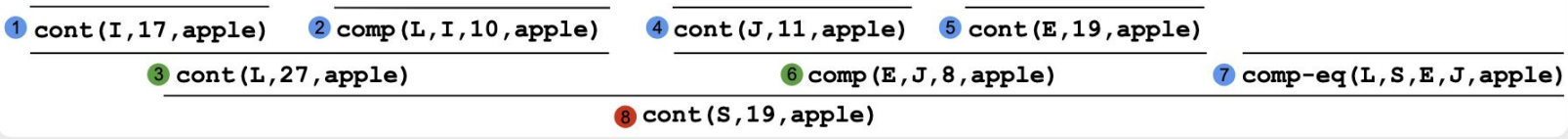
Problem specification:
- Nonlinear
- Depth: 2
- Width: 5
- Canonical ordering

Available logical form templates:

```
1. cont([agent],[quantity],[entity])
2. comp([agent1],[agent2],[quantity],[entity])
3. comp-eq([agent1],[agent2],[agent3],[agent4],[entity])
[...]
```

Available inference rules:

```
1. cont(...) comp(...) ⊢ cont(...)
2. cont(...) cont(...) ⊢ comp(...)
3. cont(...) comp(...) comp-eq(...) ⊢ cont(...)
[...]
```



Generating Problems

Step 1: Given a root logical form, sample a ***proof tree*** by iteratively applying inference rules until a stopping criterion has been reached.

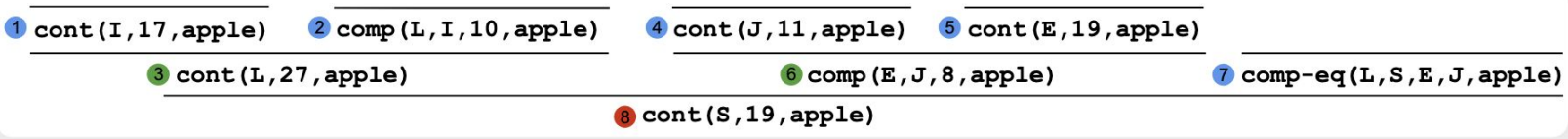
Problem specification:
Nonlinear
Depth: 2
Width: 5
Canonical ordering

Available logical form templates:

```
1. cont([agent],[quantity],[entity])
2. comp([agent1],[agent2],[quantity],[entity])
3. comp-eq([agent1],[agent2],[agent3],[agent4],[entity])
[...]
```

Available inference rules:

```
1. cont(...) comp(...) ⊢ cont(...)
2. cont(...) cont(...) ⊢ comp(...)
3. cont(...) comp(...) comp-eq(...) ⊢ cont(...)
[...]
```



Step 2: Create a ***word problem*** by mapping **leaf nodes** to text body and **root node** to a question using templates.

1 Isabella has 17 apples. 2 Lucy has 10 more apples than Isabella. 4 John has 11 apples. 5 Emily has 19 apples. 7 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

8 How many apples does Sam have?

Generating Problems

Step 1: Given a root logical form, sample a ***proof tree*** by iteratively applying inference rules until a stopping criterion has been reached.

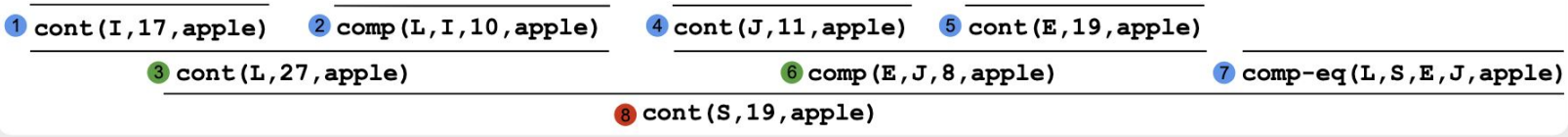
Problem specification:
Nonlinear
Depth: 2
Width: 5
Canonical ordering

Available logical form templates:

```
1. cont([agent],[quantity],[entity])
2. comp([agent1],[agent2],[quantity],[entity])
3. comp-eq([agent1],[agent2],[agent3],[agent4],[entity])
[...]
```

Available inference rules:

```
1. cont(...) comp(...) ⊢ cont(...)
2. cont(...) cont(...) ⊢ comp(...)
3. cont(...) comp(...) comp-eq(...) ⊢ cont(...)
[...]
```



Step 2: Create a ***word problem*** by mapping **leaf nodes** to text body and **root node** to a question using templates.

1 Isabella has 17 apples. 2 Lucy has 10 more apples than Isabella. 4 John has 11 apples. 5 Emily has 19 apples. 7 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily.

8 How many apples does Sam have?

Step 3: Generate a ***solution*** by mapping the nodes of the tree to proof steps. **Internal nodes** map to CoT explanations and **root node** to answer.

1 Isabella has 17 apples. 2 Lucy has 10 more apples than Isabella. 3 So Lucy has 17 + 10 = 27 apples. 4 John has 11 apples. 5 Emily has 19 apples. 6 So the difference between the number of apples John and Emily have is 8. 7 The number of apples that Lucy has more than Sam is the same as the difference between the number of apples that John has compared to Emily. 8 So Sam has 27 - 8 = 19 apples.

The MathGAP Evaluation Framework

- Can generate problems that are **arbitrarily complex**
- **Easy-to-hard** OOD generalization:
  - Easy training set
  - Complex test set
- When performance hits saturation, we can flexibly generate a new set of problems that are even more complex
  - **Dynamic benchmark**

How good are LLMs at solving increasingly complex problems?

Systematic Analysis of the Arithmetic Reasoning Capabilities of LLMs

Experiments with In-Context Learning

- Focus on in-context learning
- Can LLMs use simple problems in context to generalize to more complex ones at inference?

Experiments with In-Context Learning

- Focus on in-context learning
- Can LLMs use simple problems in context to generalize to more complex ones at inference?
- Does the distribution of in-context examples have an effect on performance?

General Experimental Setup

- For each experiment, generate multiple test sets of different degrees of complexity with 400 problems in each

General Experimental Setup

- For each experiment, generate multiple test sets of different degrees of complexity with 400 problems in each
- Four in-context distributions:
  - Zero-shot baseline
  - In-distribution baseline
  - Primitive examples: Only one proof step of the same form as in test set
  - Range of varying complexities (but simpler than test set)

General Experimental Setup

- For each experiment, generate multiple test sets of different degrees of complexity with 400 problems in each
- Four in-context distributions:
  - Zero-shot baseline
  - In-distribution baseline
  - Primitive examples: Only one proof step of the same form as in test set
  - Range of varying complexities (but simpler than test set)
- Greedy decoding, report answer accuracy

General Experimental Setup

- For each experiment, generate multiple test sets of different degrees of complexity with 400 problems in each
- Four in-context distributions:
  - Zero-shot baseline
  - In-distribution baseline
  - Primitive examples: Only one proof step of the same form as in test set
  - Range of varying complexities (but simpler than test set)
- Greedy decoding, report answer accuracy
- Models: Mixtral-8x7B, Llama3 with 8B and 70B parameters, GPT-3.5 Turbo and GPT-4o

Experiment 1: Linear Problems

- Generalization in regards to depth and width for linear problems

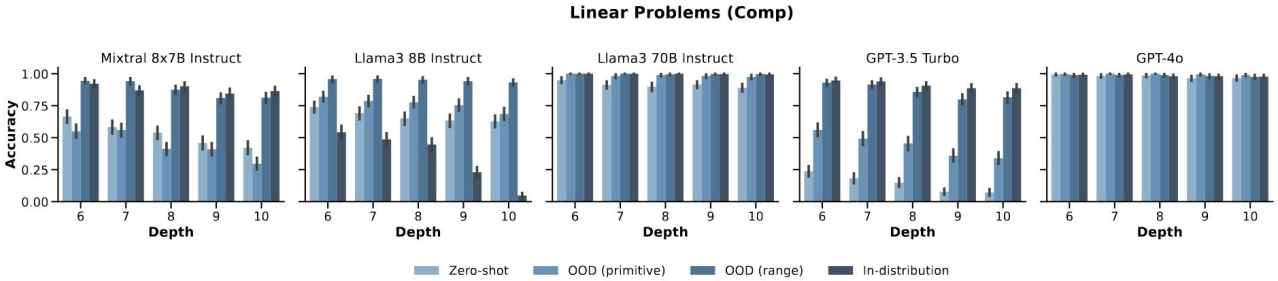
Experiment 1: Linear Problems

- Generalization in regards to depth and width for linear problems
- Three settings:
  - Depth generalization for comparison problems (Alice has 5 more apples than Bob)
  - Depth generalization for transfer problems (Alice gives 5 apples to Bob)
  - Width generalization for part-whole problems (How many apples do Alice and Bob combined?)

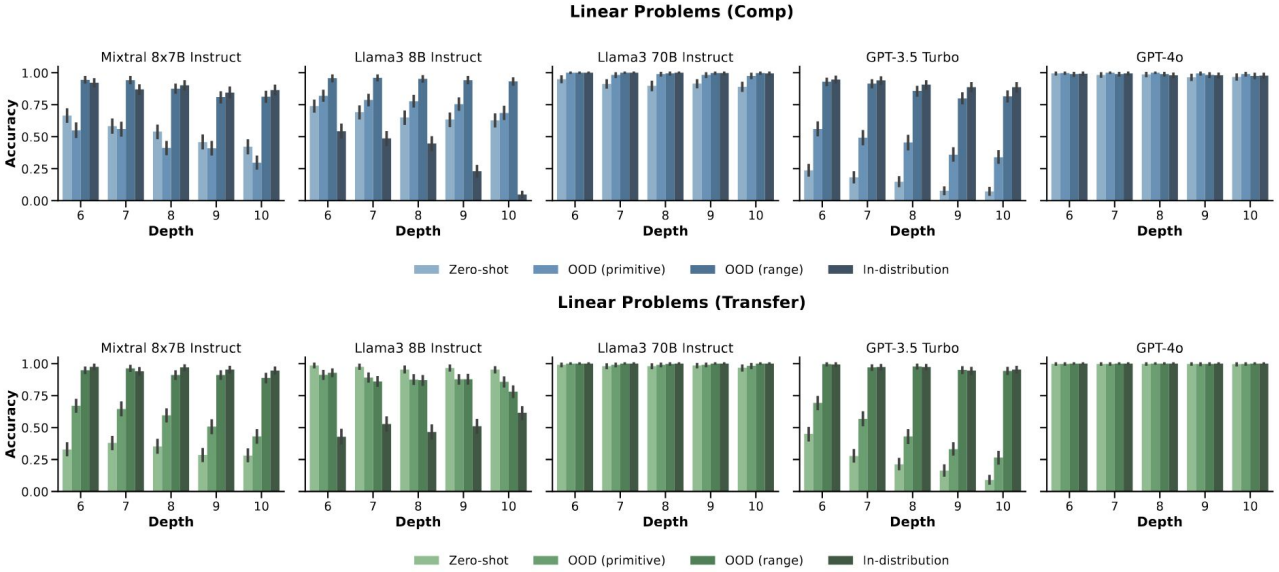
Experiment 1: Linear Problems

- Generalization in regards to depth and width for linear problems
- Three settings:
  - Depth generalization for comparison problems (Alice has 5 more apples than Bob)
  - Depth generalization for transfer problems (Alice gives 5 apples to Bob)
  - Width generalization for part-whole problems (How many apples do Alice and Bob combined?)
- Test sets:
  - Depths 6-10
  - Widths 7-11

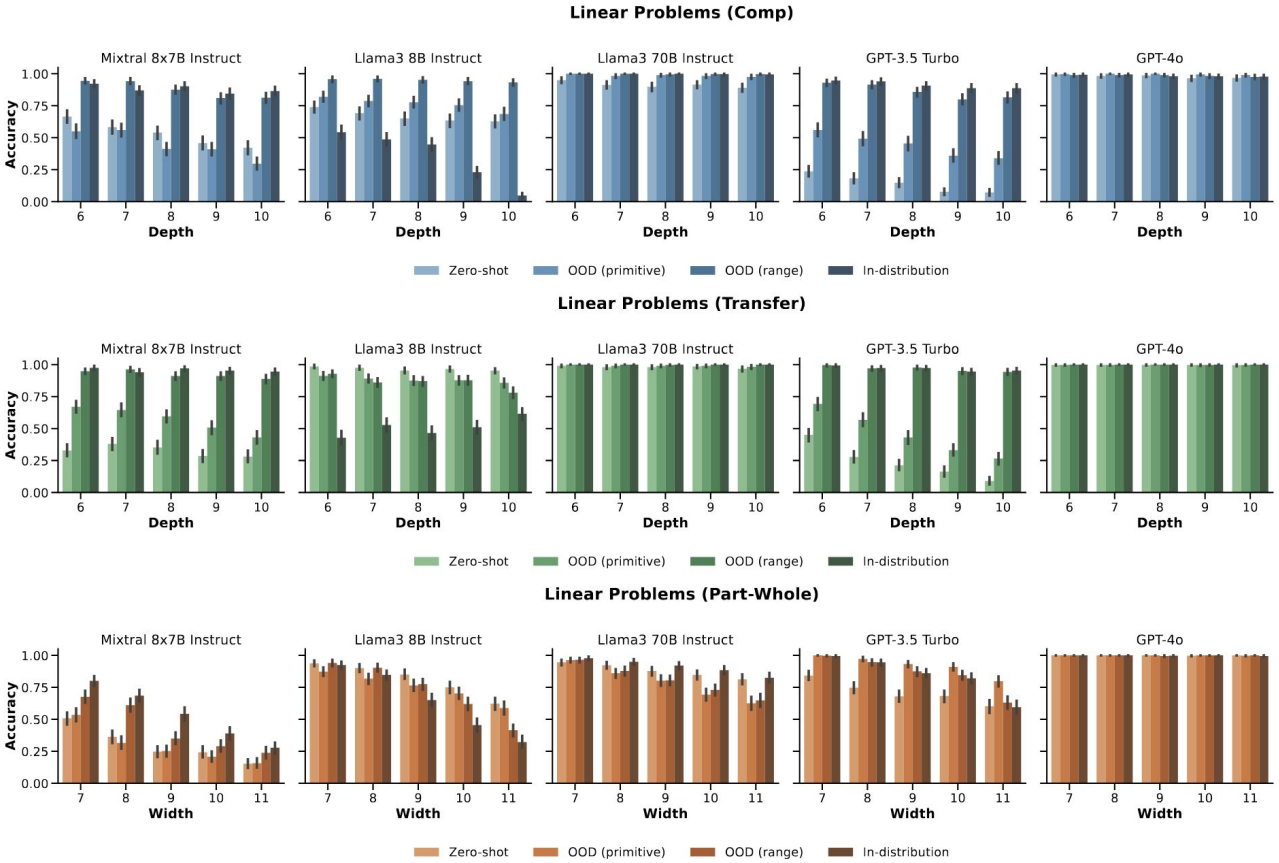
Experiment 1: Linear Problems



Experiment 1: Linear Problems



Experiment 1: Linear Problems



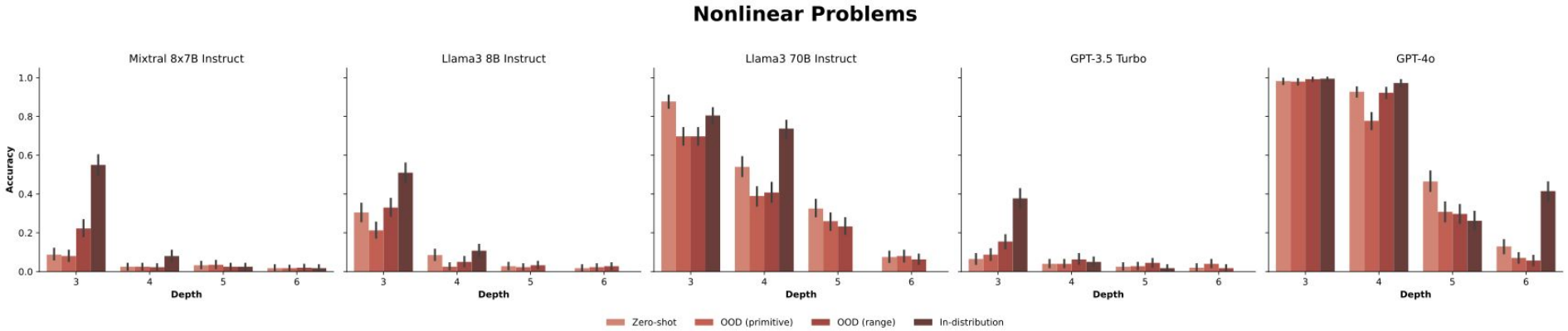
Experiment 2: Nonlinear Problems

- Generalization in regards to depth (and width) for nonlinear problems
- Nonlinear problems are generated using comparison-based inference rules

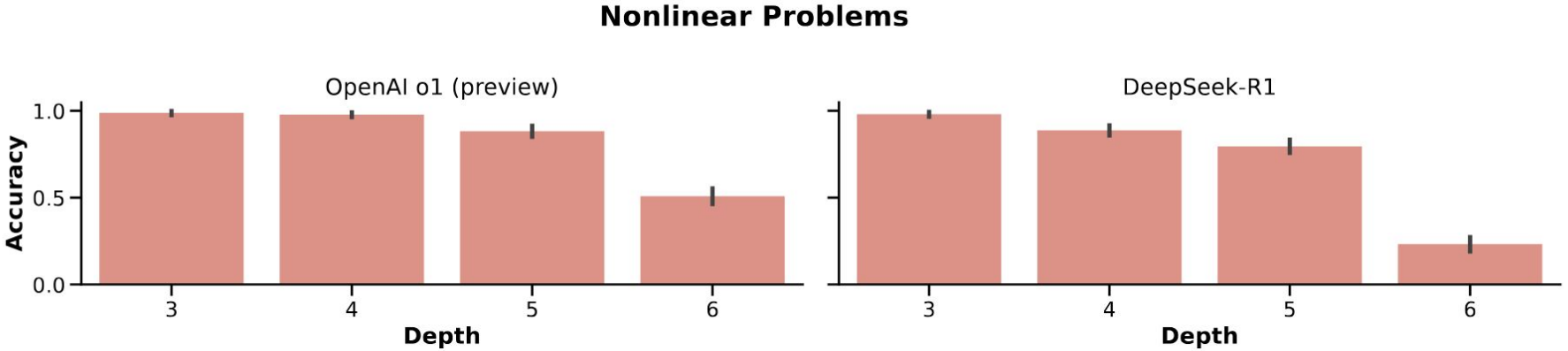
Experiment 2: Nonlinear Problems

- Generalization in regards to depth (and width) for nonlinear problems
- Nonlinear problems are generated using comparison-based inference rules
- Test sets:
  - Depths 3-6
  - Width: ~2^d for depth d

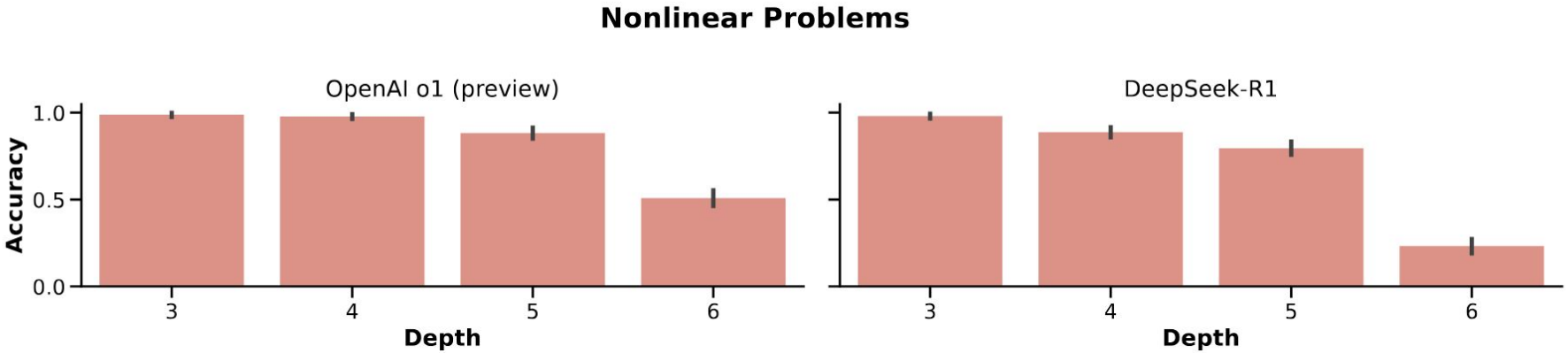
Experiment 2: Nonlinear Problems



Experiment 2: Bonus Results on o1 and R1

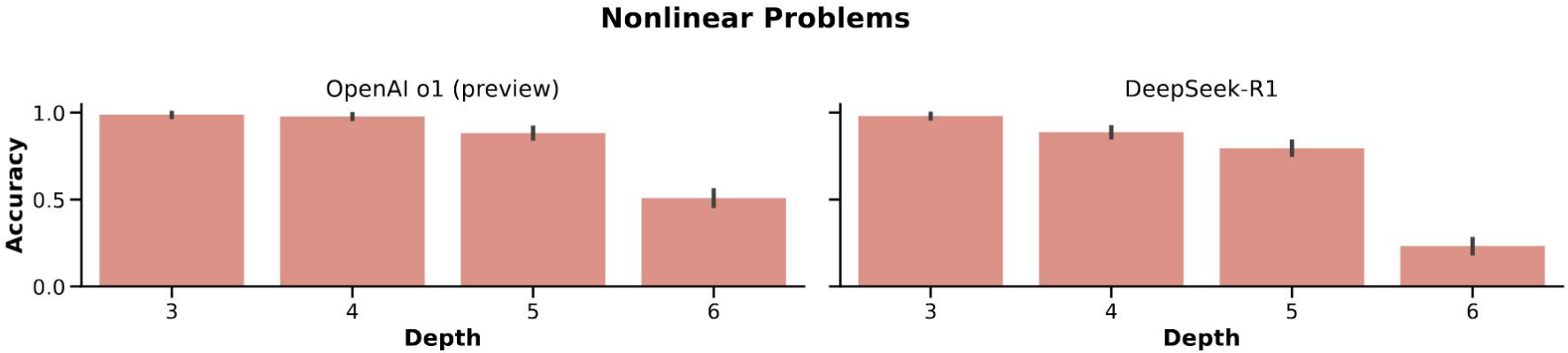


Experiment 2: Bonus Results on o1 and R1



- Depth 7: o1 performance is 0.25% with token limit 4,096; 76.5% with token limit 10,000

Experiment 2: Bonus Results on o1 and R1



- Depth 7: o1 performance is 0.25% with token limit 4,096; 76.5% with token limit 10,000
- Randomly permuted depth 7 problems (token limit 25,000): 5.0% and 11.0%

Experiment 3: Order Generalization

- LLMs are known to be sensitive to the order of axioms in reasoning (Chen et al., 2024; Eisape et al., 2024)

Experiment 3: Order Generalization

- LLMs are known to be sensitive to the order of axioms in reasoning (Chen et al., 2024; Eisape et al., 2024)
- Here: A fine-grained analysis

Experiment 3: Order Generalization

- LLMs are known to be sensitive to the order of axioms in reasoning (Chen et al., 2024; Eisape et al., 2024)
- Here: A fine-grained analysis
- Consider linear comparison problems with depth 5

Experiment 3: Order Generalization

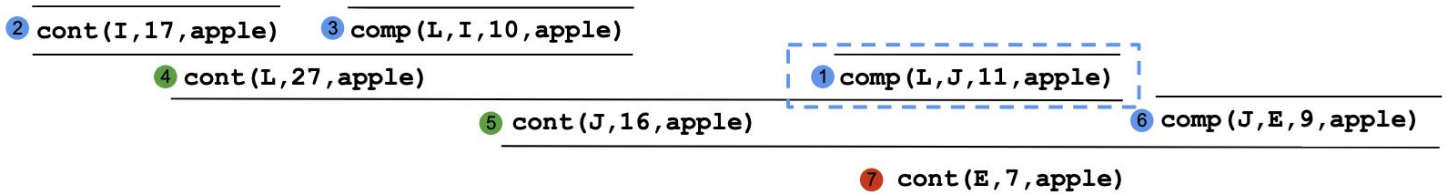
- LLMs are known to be sensitive to the order of axioms in reasoning (Chen et al., 2024; Eisape et al., 2024)
- Here: A fine-grained analysis
- Consider linear comparison problems with depth 5
- Move one sentence to the beginning of the problem

Experiment 3: Order Generalization

- LLMs are known to be sensitive to the order of axioms in reasoning (Chen et al., 2024; Eisape et al., 2024)
- Here: A fine-grained analysis
- Consider linear comparison problems with depth 5
- Move one sentence to the beginning of the problem
- Which sentences are harder to move?

Experiment 3: Order Generalization

Proof tree



Word problem (movement distance: 2)

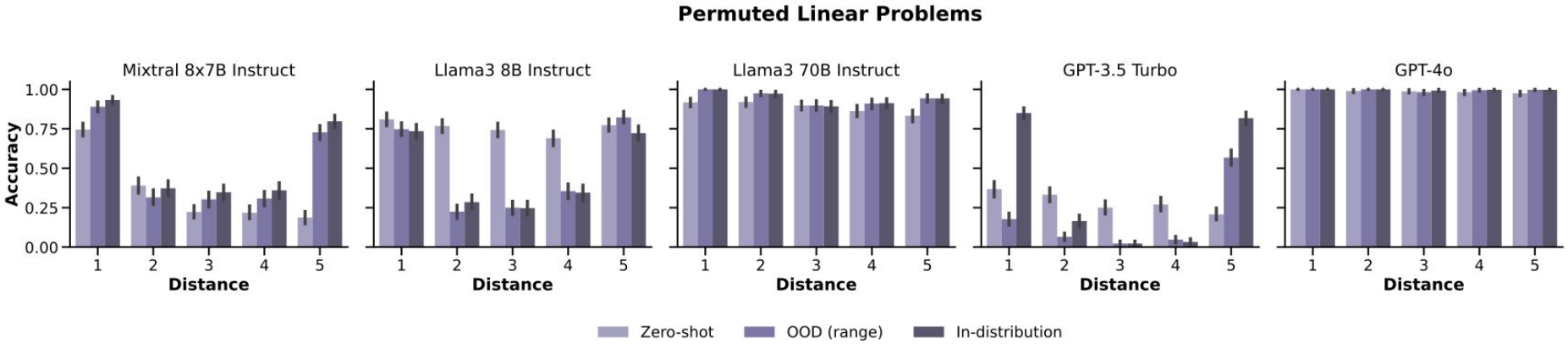
1 Lucy has 11 more apples than John. 2 Isabella has 17 apples. 3 Lucy has 10 more apples than Isabella. 6 Emily has 9 fewer apples than John.

7 How many apples does Emily have?

Chain-of-Thought Reasoning Trace

1 Lucy has 11 more apples than John. 2 Isabella has 17 apples. 3 Lucy as 10 more apples than Isabella. 4 So Lucy has 17 + 10 = 27 apples. 5 So John has 27 - 11 = 16 apples. 6 Emily has 9 fewer apples than John. 7 So Emily has 16 - 9 = 7 apples.

Experiment 3: Order Generalization



Summary

- Consistent decrease in performance as depth and width increase

Summary

- Consistent decrease in performance as depth and width increase
- But even the most complex problems are sometimes solvable, suggesting that the models are able to generalize to some extent

Summary

- Consistent decrease in performance as depth and width increase
- But even the most complex problems are sometimes solvable, suggesting that the models are able to generalize to some extent
- Nonlinear problems are more complex, even when controlling for width

Summary

- Consistent decrease in performance as depth and width increase
- But even the most complex problems are sometimes solvable, suggesting that the models are able to generalize to some extent
- Nonlinear problems are more complex, even when controlling for width
- Order permutation: Problems are harder if the sentence is moved from the middle, rather than from the beginning or end

Summary

- Consistent decrease in performance as depth and width increase
- But even the most complex problems are sometimes solvable, suggesting that the models are able to generalize to some extent
- Nonlinear problems are more complex, even when controlling for width
- Order permutation: Problems are harder if the sentence is moved from the middle, rather than from the beginning or end
- No clear relationship between in-context distribution and performance

Fin

Collaborators







Source Publications

A. Opedal, N. Stoehr, A. Saparov, M. Sachan. *World Models for Math Story Problems*. ACL 2023 (Findings).

A. Opedal*, A. Stolfo*, H. Shirakami, Y. Jiao, R. Cotterell, B. Schölkopf, A. Saparov, and M. Sachan. 2024. *Do Language Models Exhibit the Same Cognitive Biases in Problem Solving as Human Learners?* ICML 2024.

A. Opedal*, H. Shirakami*, B. Schölkopf, A. Saparov, M. Sachan. *MathGAP: Out-of-Distribution Evaluation on Problems with Arbitrarily Complex Proofs*. ICLR 2025.

Thank you for your attention!

andreas.opedal@inf.ethz.ch
X: @OpedalAndreas
Bluesky: @andreasopedal.bsky.social