



LLMs for Reasoning

Chinmay Mittal
IIT Delhi

NLP Tasks: Till Now

Sentiment Analysis

Smells Amazing ! A perfect purchase ⇒ 😊

A decent purchase ⇒ 😐

Total waste of money ⇒ 😡

Machine Translation

Le match était génial ⇒ The match was awesome

La pandémie est terminée ⇒ The pandemic has ended

Named Entity Recognition

[India]Location born [Sundar Pichai]Person is the CEO of [Google]Organization and its parent company [Alphabet]Organization

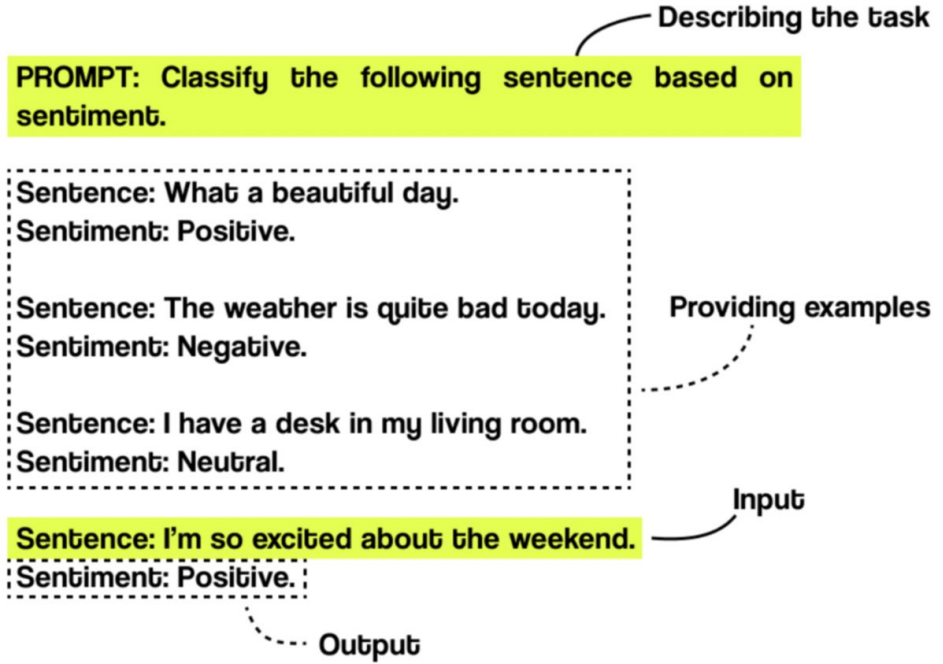
Language Tasks: Syntax & Semantics

- Meaning of Words
- Grammar
- Nuances of Language

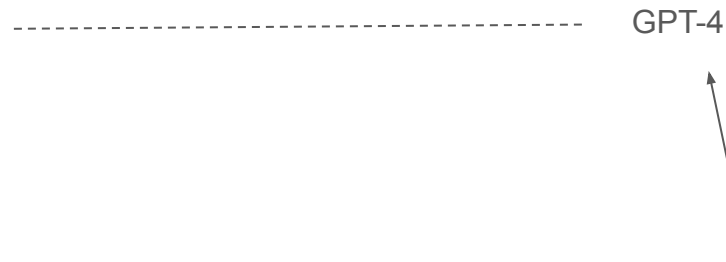
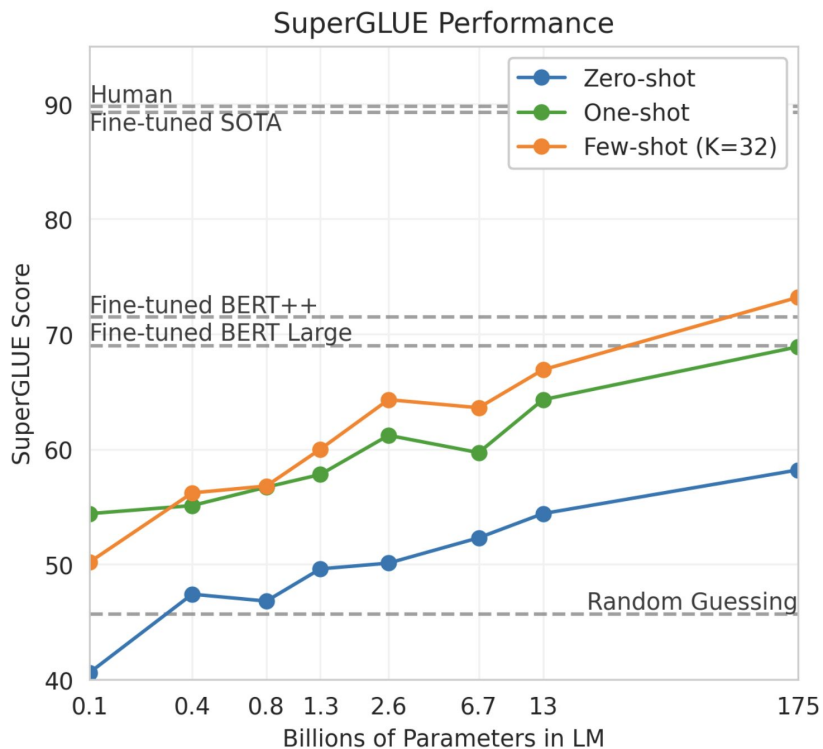
Some “Reasoning”

NLP Tasks with LLMs: Prompting

- In-context Learning
- Few Data Points
- No Fine Tuning

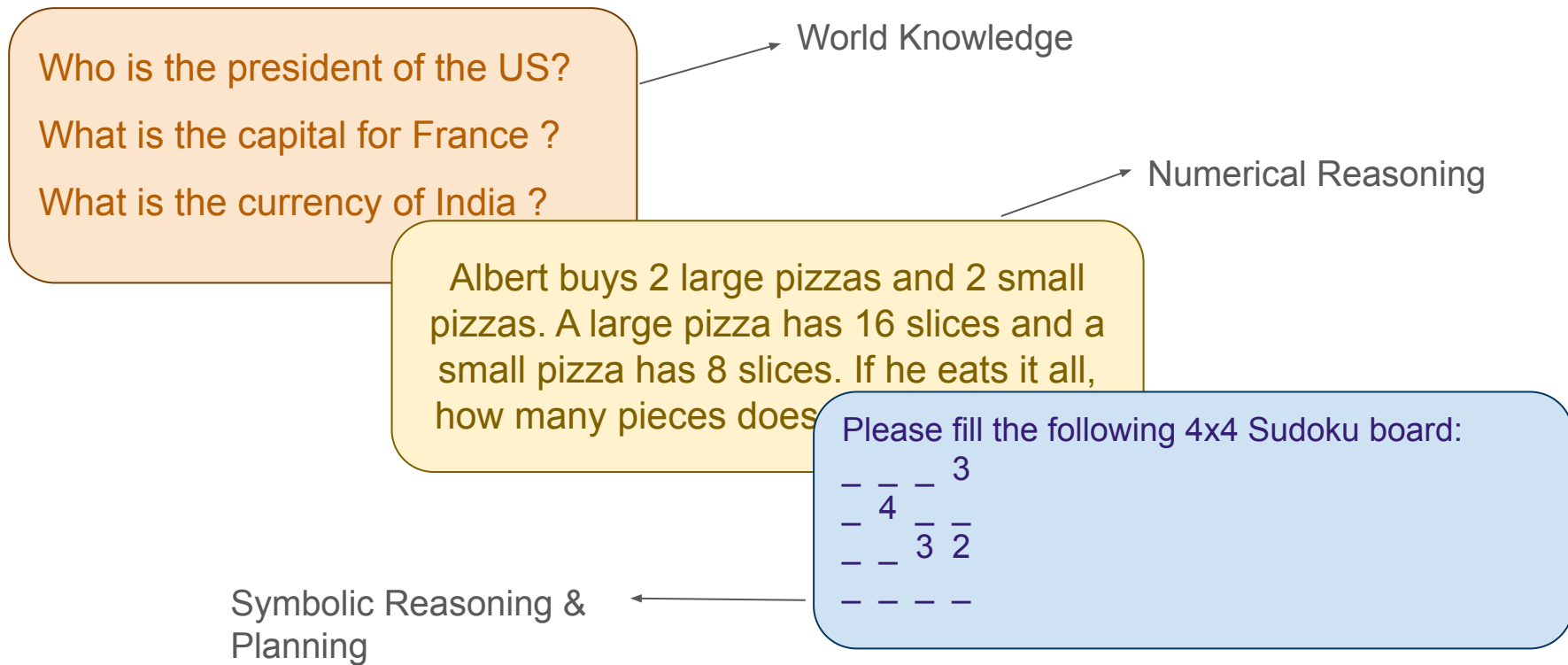


Where is NLP with LLMs ?



NLP: Beyond Language

NL \Rightarrow General QA Interface



What is Reasoning ?

Vague in Practice !

- Commonsense Reasoning
- Abductive Reasoning
- Spatial Reasoning

Extremely well defined in Theory!

- Propositional/ First-order Logic
- Geometry
- Algebra



rea·son·ing

[ˈriːz(ə)nɪŋ]

noun

the action of thinking about something in a logical, sensible way:
"he explained the reasoning behind his decision at a media conference"



Certainly! Let's delve into the fascinating world of **reasoning in Artificial Intelligence (AI)**.

Reasoning in AI refers to deriving new information from existing knowledge using logical rules and principles. It's a fundamental process that allows AI systems to make inferences, draw conclusions, and solve problems. Here are

Why Reasoning ?

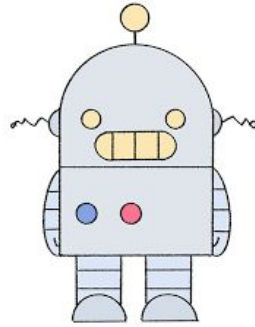
Learn from Experience

Understand the world

LLMs are stochastic parrots

Memory & Retrieval

Interact with the world



AGI

Reasoning

Planning

Can LLMs Reason ?

What are they good at ?

What are they bad at ?

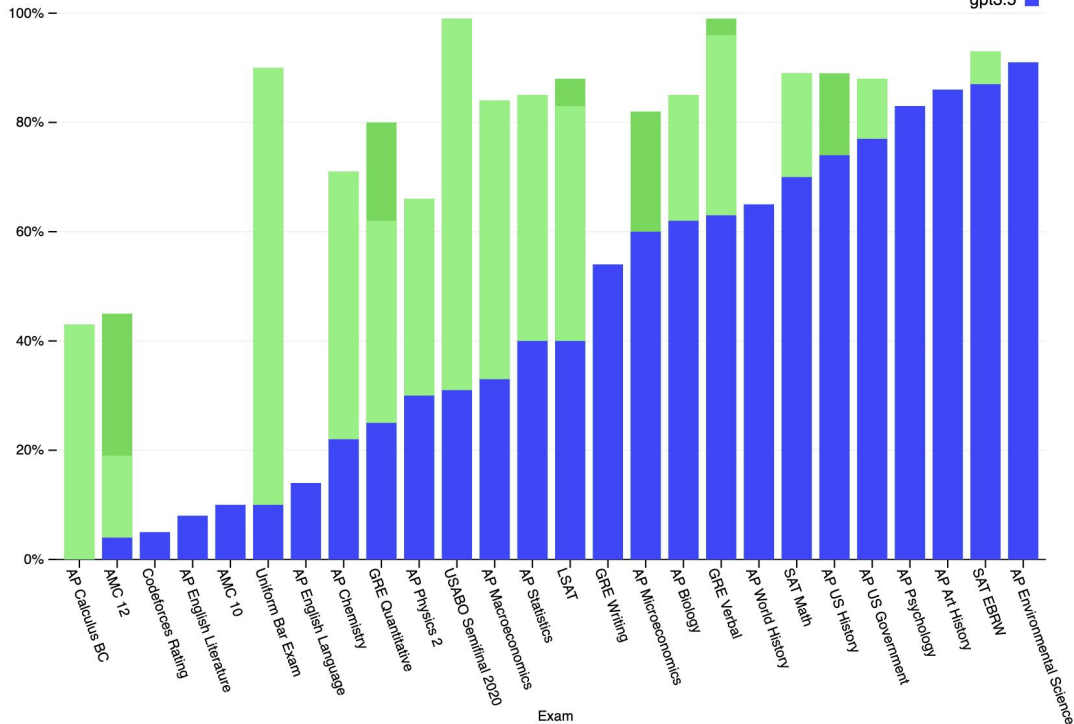
How to improve them ?

GPT-5 will achieve AGI

Reasoning with LLMs: Big Deal

Exam results (ordered by GPT-3.5 performance)

Estimated percentile lower bound (among test takers)



How good are LLMs (GPT-4) ?



- Write an essay on feminism better than an IIT Delhi student
- What is the capital of France ?
- Compute the integral of $x \sin(x)$ from 0 to 2π .
- Fill a random 9x9 Sudoku Board correctly with greater than 50% chance
- Given a shuffled deck of cards. I turn over cards until I reach the first Ace. I discard all these cards, and then turn over one more and place it face up. Which card are we more likely to see: the Ace of Diamonds, or the Queen of Clubs?
- Solve textual questions from JEE Advanced 2023 paper better than a student with 10K JEE rank

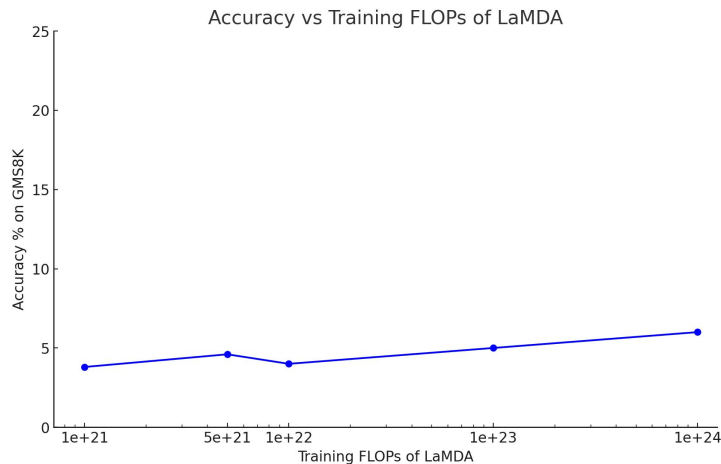
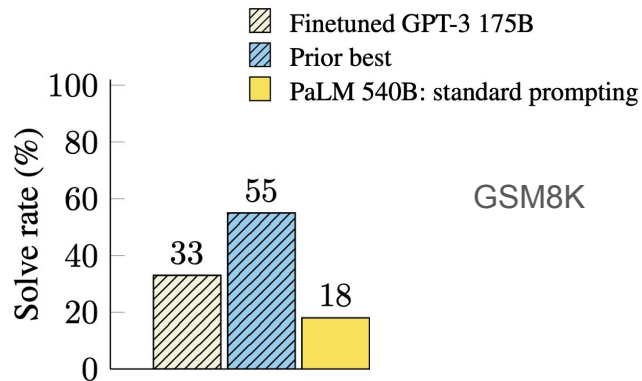
How good are LLMs (GPT-4) ?



- Write an essay on feminism better than an IIT Delhi student
- What is the capital of France ?
- Compute the integral of $x \sin(x)$ from 0 to 2π .
- Fill a random 9x9 Sudoku Board correctly with a 50% chance
- Given a shuffled deck of cards. I turn over cards until I reach the first Ace. I discard all these cards, and then turn over one more and place it face up. Which card are we more likely to see: the Ace of Diamonds, or the Queen of Clubs?
- Solve textual questions from JEE Advanced 2023 paper better better than a student with 10K JEE rank

LLMs Struggle At Reasoning

- IC learning does not work well
- Scaling models doesn't help that much !



Task	Emergence Scale
3 digit +/-	13 Billion
4-5 digit +/-	175 Billion

Why do LLMs struggle at reasoning ?

Eg. Train an LLM for the following task:

$$(x_1 \vee x_2 \vee x_6) \wedge (\neg x_1 \vee x_4 \vee x_3) \wedge (\neg x_3 \vee x_1 \vee x_7) \rightarrow T/F$$

Finding patterns to solve this task with high accuracy

eg. number of clauses / number of variables $< 4.3 \rightarrow T$

(system-1 thinking)

\neq

Knowing how to solve this task i.e. DPLL Algorithm

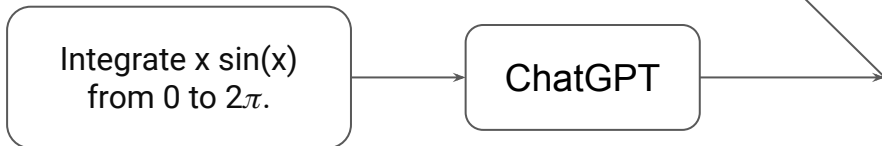
(system-2 thinking)

- At scale, can LLMs learn to perform DPLL (current LLMs, probably not)?
- What are current LLMs doing ? (probably system-1)
- Architectures / Training Tasks for moving towards system-2 type thinking

Reasoning or Retrieval ?

- LLMs are trained on enormous amounts of text ~ few 100 Gbs.
- We have very little intuition about the kinds of things the LLM has seen
- You give an LLM a question and it solves it, but has it seen the question in some form or the other is very difficult to know.
- Hence it is difficult to tell whether the LLM has learnt the first principles of reasoning or is it performing approximate pattern matching and retrieval

What about all the books on calculus that ChatGPT has seen during training ?



ChatGPT
To find the definite integral of $x \sin(x)$ from 0 to 2π , you'll need to use integration by parts. Here's how you can do it:

First, let's apply integration by parts:

$$\int u \, dv = uv - \int v \, du$$

Let $u = x$ and $dv = \sin(x) \, dx$. Then, $du = dx$ and $v = -\cos(x)$.

Now, apply the integration by parts formula:

$$\begin{aligned} \int x \sin(x) \, dx &= -x \cos(x) - \int (-\cos(x)) \, dx \\ &= -x \cos(x) + \int \cos(x) \, dx \\ &= -x \cos(x) + \sin(x) + C \end{aligned}$$

Now, evaluate the definite integral from 0 to 2π :

$$\begin{aligned} &[-x \cos(x) + \sin(x)]_0^{2\pi} \\ &= -(2\pi) \cos(2\pi) + \sin(2\pi) - (0 \cdot \cos(0) + \sin(0)) \end{aligned}$$

Since $\cos(2\pi) = 1$ and $\sin(2\pi) = 0$, and $\cos(0) = 1$ and $\sin(0) = 0$, the expression simplifies to:

$$= -2\pi$$

So, the value of the definite integral of $x \sin(x)$ from 0 to 2π is -2π .

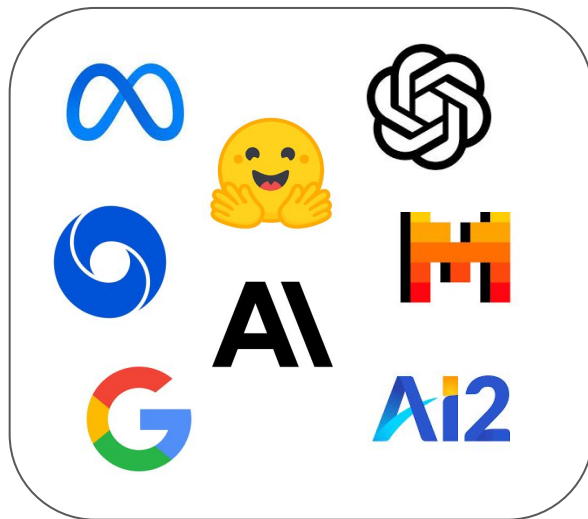
How to improve LLMs at Reasoning ?

Knob-1: Prompting

What do I ask the LLM to do ?



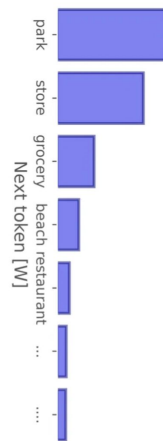
Reasoning Question



Black Box LLM

Knob-2: Decoding

How do I extract the answer from the LLM ?



Next Token Probability

Where does Standard Prompting Fail ?

- How do we think about these problems ?
 - Problem **Understanding**
 - **Decomposition** to Intermediate Steps
 - **Solve** Intermediate Steps to get Final Answer

How do I get the model to do this ?

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. ✓

Doing Multi Step Reasoning in a Single Pass is Challenging !

Model Input

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?

Model Output

A: The answer is 27. ✗

Chain of Thought Prompting

- Augment Exemplars with the Chain of Thought to get to the answer
- Chain of Thought is the sequence of intermediate reasoning steps
- LLMs learn to do Chain of Thought reasoning via in-context learning

Model Input

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: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. 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?

Model Output

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. ✓

CoT Intuition

LLM pre training essentially stores training data as a compressed index of some sort in their weights

In-context examples guide the LLM to retrieve relevant pre-training data from the weights

In context exemplars guide the LLM to leverage pretraining data to produce CoT like output

Producing each token is much easier and can be done using pre-training data

LLM can produce more token for problems requiring more computation

Two tasks: Produce CoT like output & make sure the output is correct

Model Input

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: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. 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?

Model Output

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. ✓

CoT Intuition

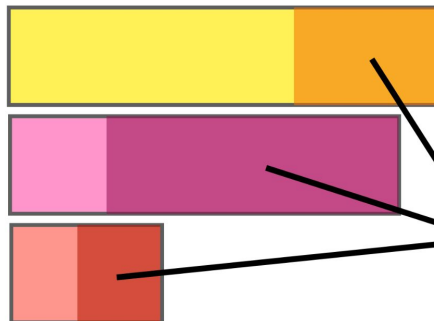
- Instead of solving the problem in a single pass, the model produces multiple tokens
- While producing an intermediate sequence of tokens the LLM is doing a particular task for eg. numerical reasoning
- With scale, LLMs get better at all of these tasks and hence better at the overall task reaching the correct answer

Types of errors made by a 62B language model:

Semantic understanding
(62B made 20 errors of this type,
540B fixes 6 of them)

One step missing
(62B made 18 errors of this type,
540B fixes 12 of them)

Other
(62B made 7 errors of this type,
540B fixes 4 of them)



Errors fixed by
scaling from
62B to 540B

CoT: Explainability

Question

Tom's ship can travel at 10 miles per hour. He is sailing from 1 to 4 PM. He then travels back at a rate of 6 mph. How long does it take him to get back?

62B Model Output

He travels from 1 to 4 PM. That means he travels for 3 hours. He travels back at 6 mph. That means he travels for $3 * 6 = 18$ miles. The answer is 18. **(Incorrect: semantic understanding error)**

540B Model Output

He travels at 10 mph from 1 to 4 PM. This means he travels 3 hours. 3 hours at 10 mph means he travels $3 * 10 = 30$ miles. He then travels back at 6 mph. This means he travels 6 miles per hour. He has to travel 30 miles, so it takes him $30 / 6 = 5$ hours. The answer is 5. **(Correct)**

CoT: Wide Applicability

Math Word Problems (free response)

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: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Math Word Problems (multiple choice)

Q: How many keystrokes are needed to type the numbers from 1 to 500? Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788

A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500. $9 + 90(2) + 401(3) = 1392$. The answer is (b).

Arithmetic Reasoning

Last Letter Concatenation

Q: Take the last letters of the words in "Lady Gaga" and concatenate them.

A: The last letter of "Lady" is "y". The last letter of "Gaga" is "a". Concatenating them is "ya". So the answer is ya.

Coin Flip (state tracking)

Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up?

A: The coin was flipped by Maybelle. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.

Symbolic Reasoning

Sports Understanding

Q: Is the following sentence plausible? "Joao Moutinho caught the screen pass in the NFC championship."

A: Joao Moutinho is a soccer player. The NFC championship is part of American football, not soccer. So the answer is no.

StrategyQA

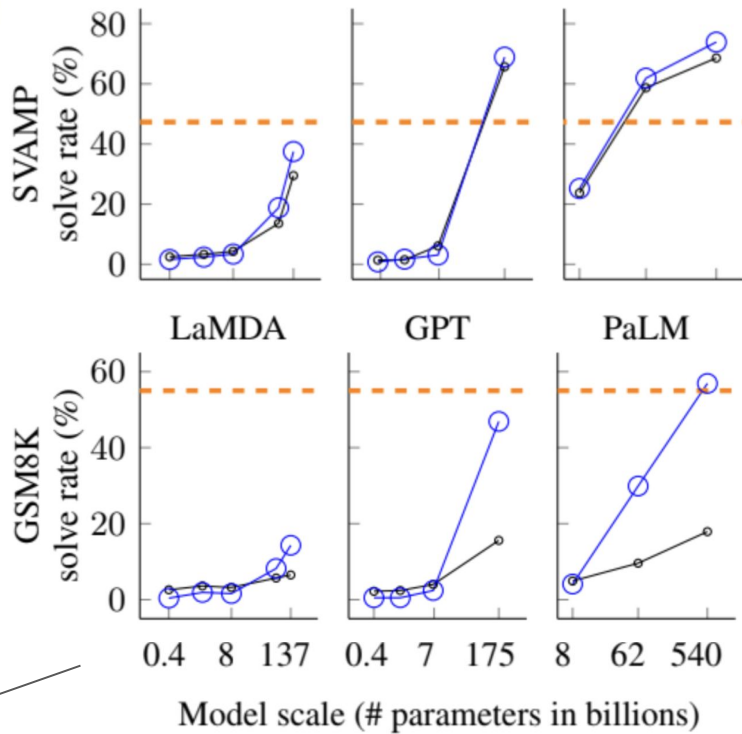
Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm^3 , which is less than water. Thus, a pear would float. So the answer is no.

CommonSense Reasoning

CoT: Huge Performance Gains

- Standard prompting
- Chain-of-thought prompting
- - - Prior supervised best



Huge Gains on Challenging datasets !

Closed to SoTA !

Emergent Ability !

CoT: Pros ?

- Variable Computation
- Explainability
- Wide Applicability
- No Training
- Huge Performance Gains

Model Input

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: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. 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?

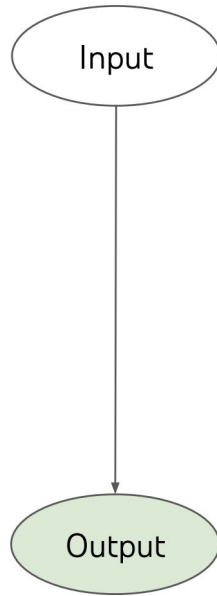
Model Output

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. ✓

What have we achieved ?

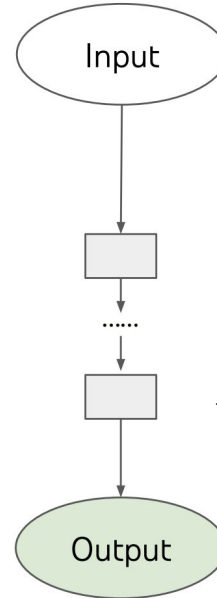


Often Multiple ways to reach the answer !



Standard Prompting

Guide the LLM through IC learning
to perform explicit reasoning



CoT prompting

What if LLM gets something wrong ?



Decoding with LLMs: Greedy Decoding



Question
+
Prompt

The probability distribution predicted by the LLM is a way to explore diverse reasoning paths !

model's vocabulary

0.298	0.213	0.125	0.098	0.087	0.056	0.013	0.011
the	their	another	a	an	victory	its	all

Select token w/
highest
probability

'the'



Self Consistency Decoding

CoT



This means she uses $3 + 4 = 7$ eggs every day. She sells the remainder for \$2 per egg, so in total she sells $7 * \$2 = \14 per day. The answer is \$14.

Greedy Decoding

SC

Temperature Sampling to sample Diverse Reasoning Paths !

Q: Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder for \$2 per egg. How much does she make every day?



She has $16 - 3 - 4 = 9$ eggs left. The answer is 18
So she makes $\$2 * 9 = \18 per day.

This means she she sells the remainder for $\$2 * (16 - 4 - 3) = \26 per day. The answer is 26

She eats 3 for breakfast, so she has $16 - 3 = 13$ left. Then she bakes muffins, so she has $13 - 4 = 9$ eggs left. So she has $9 \text{ eggs} * \$2 = \18 . The answer is 18

Aggregate Answers

The answer is 18

Marginalize out reasoning paths !

Self Consistency

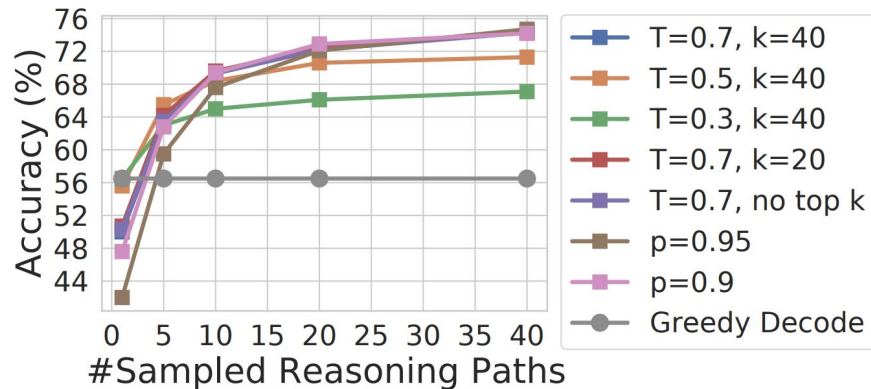
Pros

- Simple and Effective
- No Training / Fine Tuning
- Model Calibration / Confidence

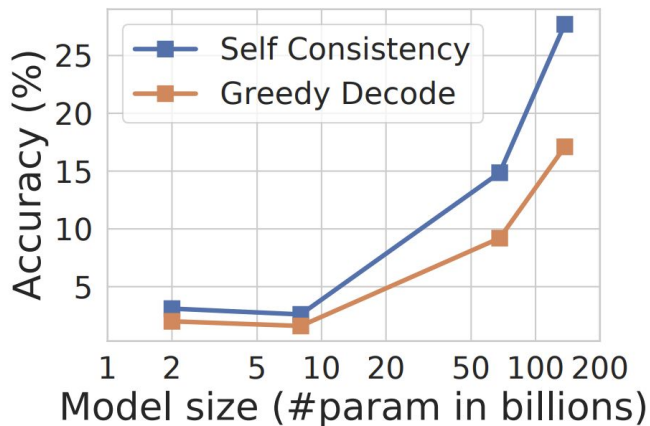
Cons

- Computation Cost
- Does not fix the inherent reasoning limitations of the LLM

GSM8K with PaLM- 540 B



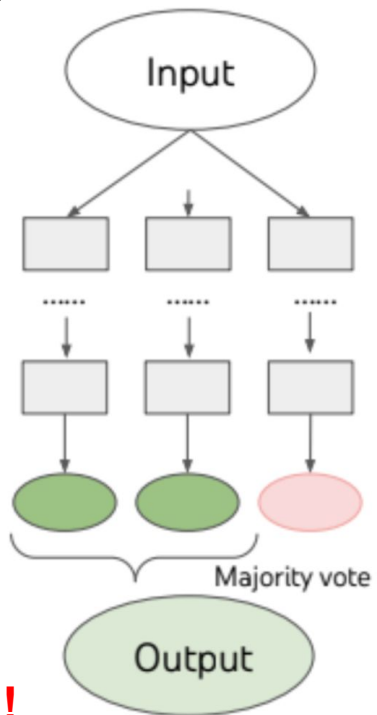
GSM8K with LaMDA



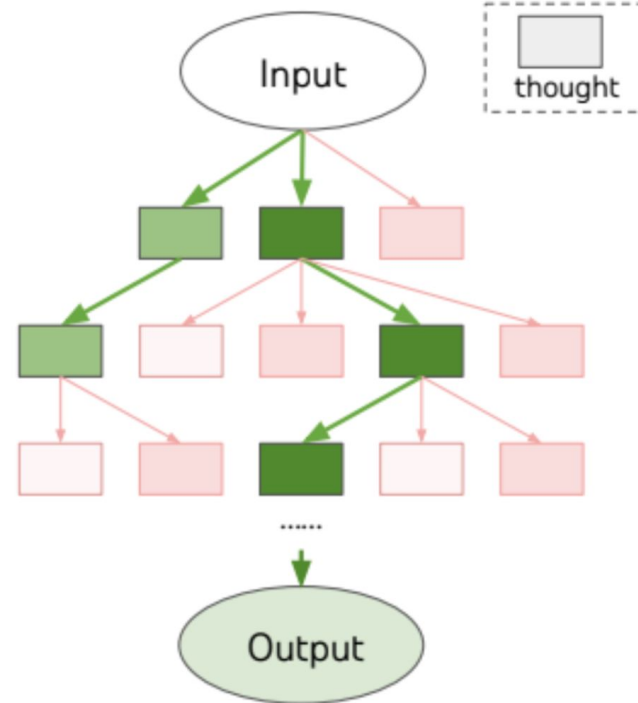
What next CoT + SC → ToT ?

General Problem Solver ?

		3	
		1	
			1
3		2	



Special Case



Reasoning = Search !

Tree of Thoughts Prompting

Search Recap: ToT setup

LLM + Classical AI

State

Partial Solution
Input + Sequence of Thoughts

Evaluation

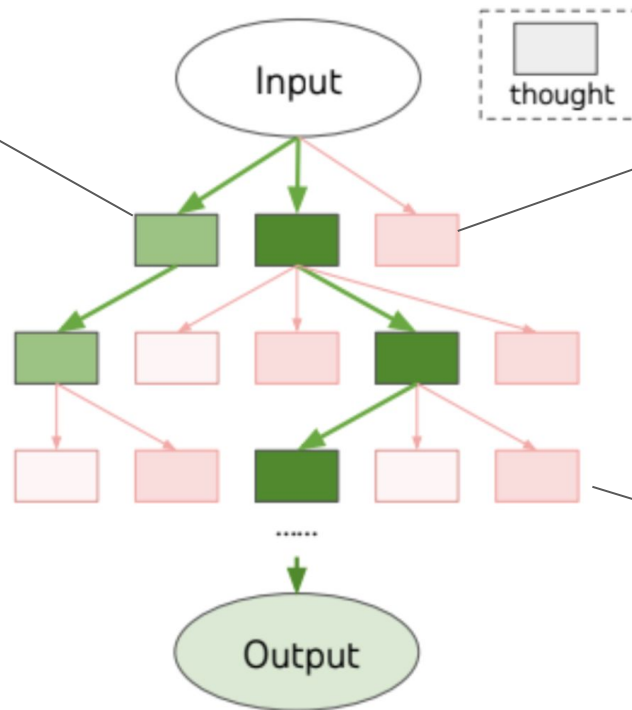
Heuristic Function
How good is this state

Action

What to do next ?
Next Thought

Search Strategy

How should we grow the tree ?
BFS/DFS/A*/MCTS



State: Example Crosswords

- Partial Solution i.e. $s = [x, z_1, \dots, z_n]$
- Input State + Sequence of Thoughts till now

t	a	s	k	s
m	o	t	o	r
s	a	l	o	n

Horizontal

h1: something to be done: tasks

h2: an engine: motor

...

Vertical

v1: To heap: tm_s_

v3: Pretentious; Flowery: st_l_

v5: Desiccator; more dry: sr_n_

Horizontal

h1: something to be done

h2: an engine

...

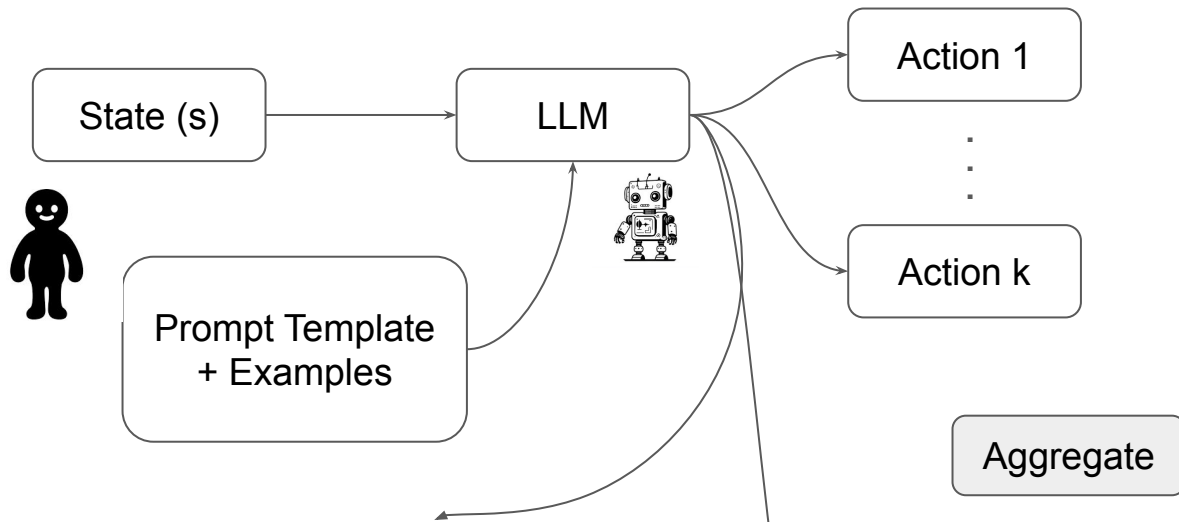
Vertical

v1: To heap

v3: Pretentious; Flowery

v5: Desiccator; more dry

Actions: Thought Generator $G(p_{\theta}, s, k)$



Horizontal

h1: something to be done: tasks

h2: an engine: motor

h4: place where a barber works: _____

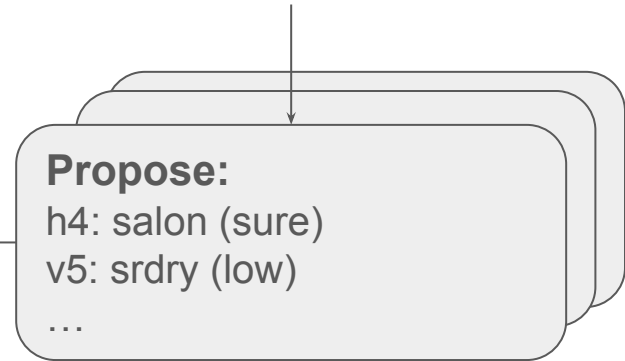
...

Vertical

v1: To heap: tm_s_

v3: Pretentious; Flowery: st_l_

v5: Desiccator; more dry: sr_n_



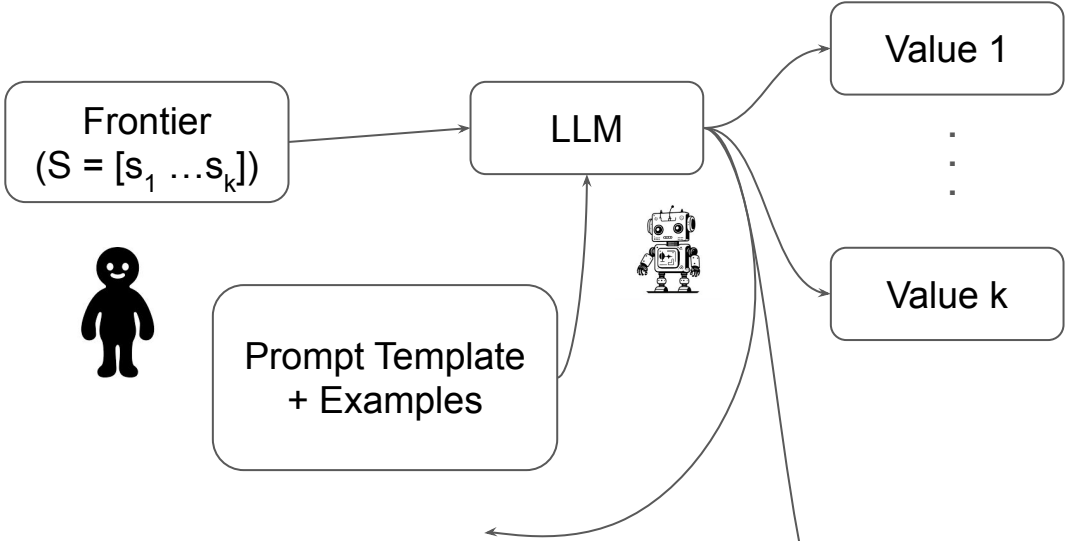
Sample k times

$$z_{i+1} \sim p^{\text{CoT}}_{\text{LLM}}(z_{i+1} | s = x, z_1 \dots z_i)$$

Propose k actions in one pass

$$[z^{(1)} \dots z^{(k)}] \sim p^{\text{propose}}_{\text{LLM}}(z_{i+1}^{(1\dots k)} | s)$$

State Frontier **Evaluator**: $V(p_\theta, S)$



Horizontal
 h4: place where a barber works: _____
(sure)
 ...

Vertical
 v1: To heap: tm_s_
(impossible)
 v5: Desiccator; more dry: sr_n_
(maybe)

Value: Categorical/Real Number Score

$$V(p_{LLM}, S)(s) = p_{LLM}^{value}(v|s) \quad \forall s \in S$$

Note: Compare States explicitly

Not Programmed like Classical AI => No Manual Intervention
 Not Learnt Like AlphaGo => No Training

Search Algorithm

Algorithm 1 ToT-BFS($x, p_\theta, G, k, V, T, b$)

Require: Input x , LM p_θ , thought generator $G()$ & size limit k , states evaluator $V()$, step limit T , breadth limit b .

$S_0 \leftarrow \{x\}$

for $t = 1, \dots, T$ **do**

$S'_t \leftarrow \{[s, z] \mid s \in S_{t-1}, z_t \in G(p_\theta, s, k)\}$

$V_t \leftarrow V(p_\theta, S'_t)$

$S_t \leftarrow \arg \max_{S \subset S'_t, |S|=b} \sum_{s \in S} V_t(s)$

end for

return $G(p_\theta, \arg \max_{s \in S_T} V_T(s), 1)$

Algorithm 2 ToT-DFS($s, t, p_\theta, G, k, V, T, v_{th}$)

Require: Current state s , step t , LM p_θ , thought generator $G()$ and size limit k , states evaluator $V()$, step limit T , threshold v_{th}

if $t > T$ **then** record output $G(p_\theta, s, 1)$

end if

for $s' \in G(p_\theta, s, k)$ **do** ▷ sorted candidates

if $V(p_\theta, \{s'\})(s) > v_{thres}$ **then** ▷ pruning

 DFS($s', t + 1$)

end if

end for

ToT

Pros

- Towards General Problem Solving with LLMs
- Performance Gains
- Interpretability
- Classical AI + LLMs
- Modular

Cons

- Computationally Expensive
- Task Specific Prompt Engineering
- Not Needed for many tasks

Neverending list of Prompting Techniques

What is the best way to guide the LLM to the correct answer ?

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

Jason Wei
Brian Ichter

Tree of Thoughts: Deliberate Problem Solving with Large Language Models

Shunyu Yao
Princeton Unive

Progressive-Hint Prompting Improves Reasoning in Large Language Models

Thomas L.
Princeton U

Chuanyu
C
{cyzheng21, liyu}
h

Large Language Models are Zero-Shot Reasoners

T
t.kojima

Machel R
Google Res

LEAST-TO-MOST PROMPTING ENABLES COMPLEX REASONING IN LARGE LANGUAGE MODELS

Denny Zhou^{†*} Nathanael Schärli[†] Le Hou[†] Jason Wei[†] Nathan Scales[†] Xuezhi Wang[†]

Dale Schuurmans[†] Claire Cui[†] Olivier Bousquet[†] Quoc Le[†] Ed Chi[†]

[†] Google Research, Brain Team

How to improve LLMs at Reasoning ?

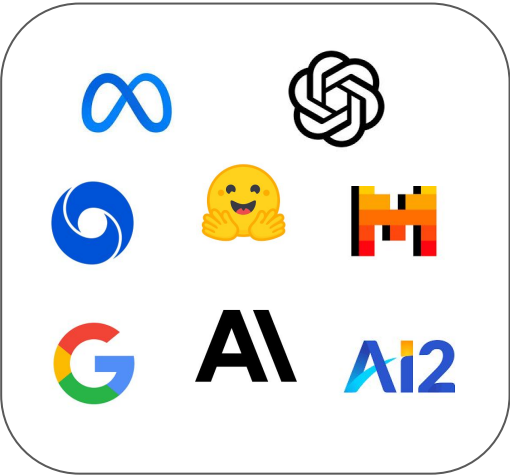
LLM Modulo Frameworks



Prompting



Reasoning Question



Black Box LLM

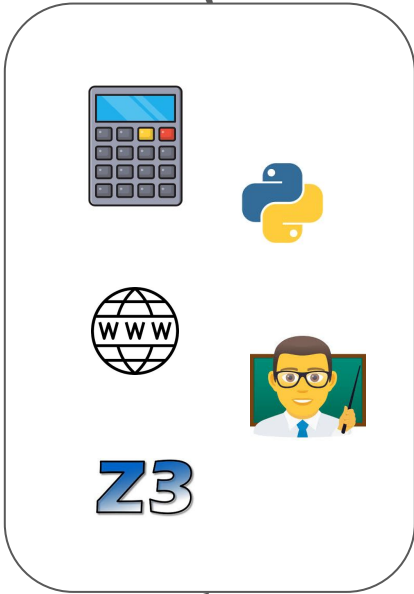
I am **GOOD** at 😊

- Natural Language Understanding !
- Language Translation !
-

I am **BAD** at / costly for 😞

- Maths
- Reasoning
-

External Agent



Decoding

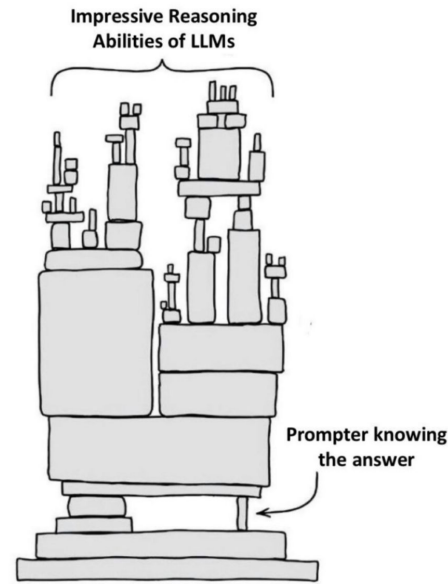
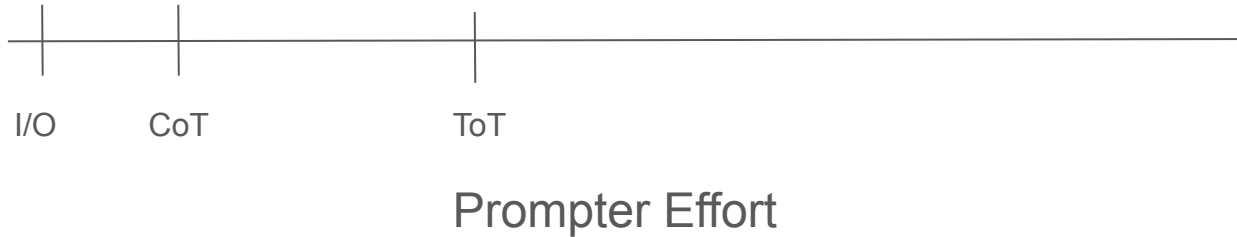


Next Token Probability

I do what I am good at, the agent does what it is good at

Prompting: Prompter as Tool

- LLMs are dream machines, and our prompt guides their dreams.
- We want that the LLM to end up in factually correct territory
- All prompting methods can be seen as techniques to guide the LLM to stay away from the factually incorrect territory
- But there is a risk of answer leakage from the prompter: In the effort of guiding the LLM are we revealing the answer ?

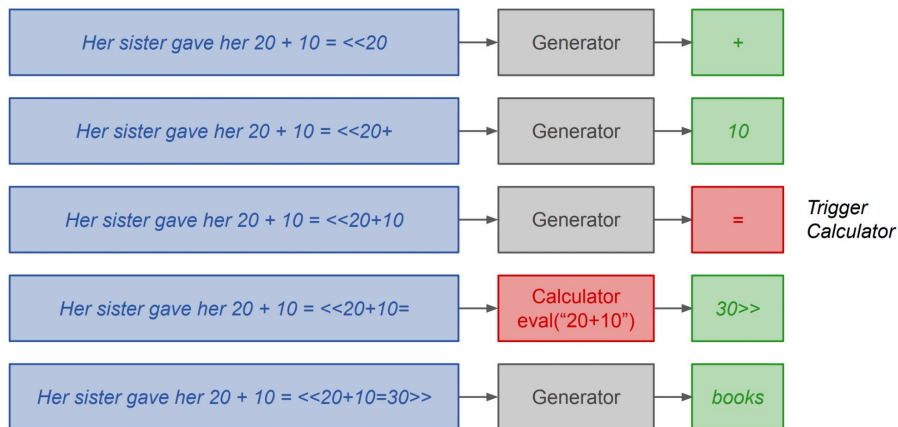


Fixing Numerical Reasoning Mistakes

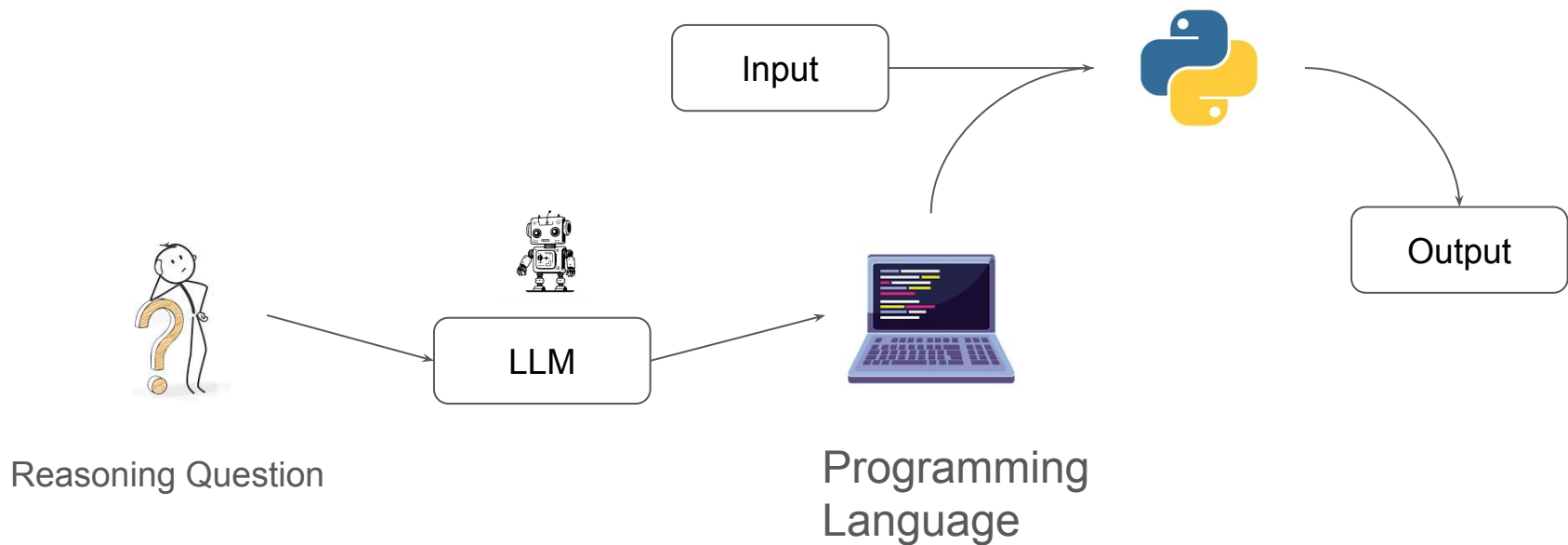
QUESTION: Dan plants 3 rose bushes. Each rose bush has 25 roses. Each rose has 8 thorns. How many thorns are there total?

MODEL ANSWER (INCORRECT; CALCULATOR ERROR ONLY): Dan plants 3 rose bushes. Each rose bush has 25 roses. Each rose has 8 thorns. So $3 \times 25 \times 8 = 300$. The answer is 300. ✘

└─ Ask a calculator to do this ! But How ?



Program Interpreters as Calculators



Leverage LLMs Code Writing Abilities !



LLMs & Code

- LLMs are pre-trained on code
- LLMs do well at coding tasks

Total dataset size = 780 billion tokens	
Data source	Proportion of data
Social media conversations (multilingual)	50%
Filtered webpages (multilingual)	27%
Books (English)	13%
GitHub (code)	5%
Wikipedia (multilingual)	4%
News (English)	1%

PaLM 540 B training code

		Pretraining only		Code Finetuning			
		LaMDA 137B	PaLM 540B	Codex 12B ^a	Davinci Codex*	PaLM Coder 540B	
HumanEval ⁽⁰⁾	pass@100	47.3	76.2	72.3	81.7	88.4	→ Text to Code
MBPP ⁽³⁾	pass@80	62.4 ^b	75.0	–	84.4	80.8	
TransCoder ⁽³⁾	pass@25	–	79.8	–	71.7	82.5	
HumanEval ⁽⁰⁾	pass@1	14.0	26.2	28.8	36.0	36.0	→ Code Translation
MBPP ⁽³⁾	pass@1	14.8 ^b	36.8	–	50.4	47.0	
GSM8K-Python ⁽⁴⁾	pass@1	7.6	51.3	–	32.1	50.9	
TransCoder ⁽³⁾	pass@1	30.2	51.8	–	54.4	55.1	
DeepFix ⁽²⁾	pass@1	4.3	73.7	–	81.1	82.1	

Program Aided Language Models

LLMs can Think in Code !



PAL

CoT

Input

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: Roger started with 5 tennis balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The bakers at the Beverly Hills Bakery baked 200 loaves of bread on Monday morning. They sold 93 loaves in the morning and 39 loaves in the afternoon. A grocery store returned 6 unsold loaves. How many loaves of bread did they have left?

Model Output

A: The bakers started with 200 loaves. They sold 93 in the morning and 39 in the afternoon. So they sold $93 + 39 = 132$ loaves. The grocery store returned 6 loaves. So they had $200 - 132 - 6 = 62$ loaves left. The answer is 62.



Input

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: Roger started with 5 tennis balls.
`tennis_balls = 5`
2 cans of 3 tennis balls each is
`bought_balls = 2 * 3`
tennis balls. The answer is
`answer = tennis_balls + bought_balls`

Q: The bakers at the Beverly Hills Bakery baked 200 loaves of bread on Monday morning. They sold 93 loaves in the morning and 39 loaves in the afternoon. A grocery store returned 6 unsold loaves. How many loaves of bread did they have left?

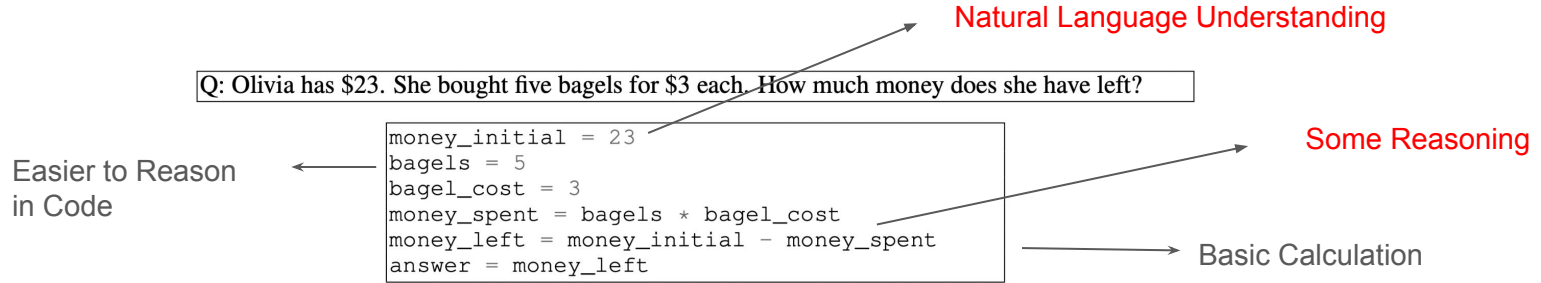
Model Output

A: The bakers started with 200 loaves
`loaves_baked = 200`
They sold 93 in the morning and 39 in the afternoon
`loaves_sold_morning = 93`
`loaves_sold_afternoon = 39`
The grocery store returned 6 loaves.
`loaves_returned = 6`
The answer is
`answer = loaves_baked - loaves_sold_morning`
`- loaves_sold_afternoon + loaves_returned`

```
>>> print(answer)  
74
```



Offloading Reasoning to the Interpreter



Q: On the table, you see a bunch of objects arranged in a row: a purple paperclip, a pink stress ball, a brown keychain, a green scrunchiephone charger, a mauve fidget spinner, and a burgundy pen. What is the color of the object directly to the right of the stress ball?

```
...
stress_ball_idx = None
for i, object in enumerate(objects):
    if object[0] == 'stress ball':
        stress_ball_idx = i
        break
# Find the directly right object
direct_right = objects[stress_ball_idx+1]
# Check the directly right object's color
answer = direct_right[1]
```


Looping & Conditionals

Offloading Reasoning to the Interpreter

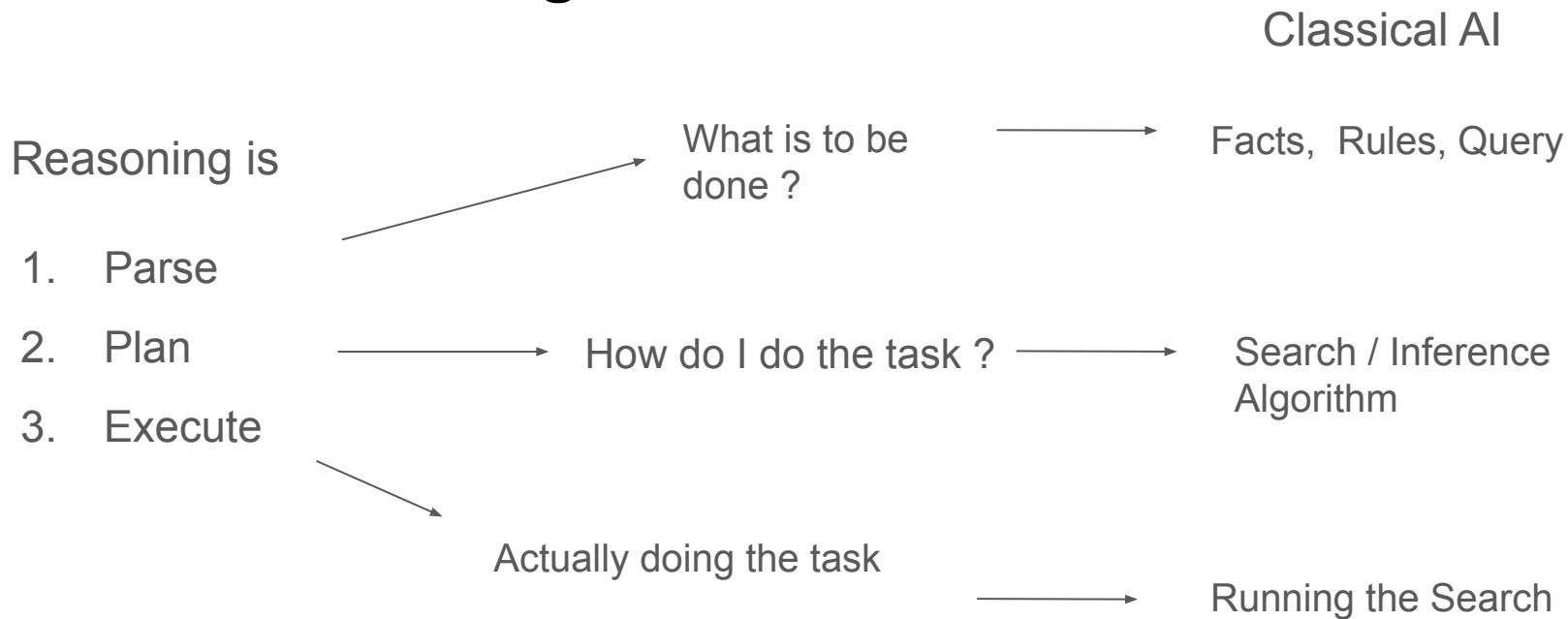
Q: I have a chair, two potatoes, a cauliflower, a lettuce head, two tables, a cabbage, two onions, and three fridges. How many vegetables do I have?

```
# note: I'm not counting the chair, tables,  
or fridges  
vegetables_to_count = {  
    'potato': 2,  
    'cauliflower': 1,  
    'lettuce head': 1,  
    'cabbage': 1,  
    'onion': 2  
}  
answer = sum(vegetables_to_count.values())
```

Python as a
powerful API



What is Reasoning ?



CoT ?

Reasoning is

1. Parse / Understand
2. Plan / Search
3. Execute

Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

CoT

LLM (Parse + Plan + Execute)



Let X be the number of Pokemon Stan has caught. Alex has caught 5 more than Stan, so Alex has caught $X + 5$.

Stan has caught 13 less than 4 times as many as Adelwolfe has caught, so Stan has caught $4X - 13$. Together they have caught 339 Pokemon, so $X + 5 + 4X - 13 = 339$.

Combining like terms produces $5X + 5 = 339$. Subtracting 5 from both sides produces $5X = 334$. Dividing both sides by 5 produces $X = \$66.80$, so Stan has caught 66 Pokemon.

The answer is 66. ❌

PAL ?

Reasoning is

1. **Parse**

2. **Plan** → LLM not playing to its strengths ! 

3. **Execute**

Imperative Specification ←

Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

ProgramLM

LLM (Parse + Plan)

```
total_pokemon = 339
alex_pokemon = 5
stan_pokemon = 4
adelwolfe_pokemon = 13
stan_pokemon = (total_pokemon -
alex_pokemon - adelwolfe_pokemon *
stan_pokemon) / (1 - stan_pokemon)
result = stan_pokemon
```



Classical AI to the Rescue



Natural Language Reasoning Question



Parse



Symbolic Representation of Facts / Rules



Sophisticated Reasoning / Planning Algorithms

Plan

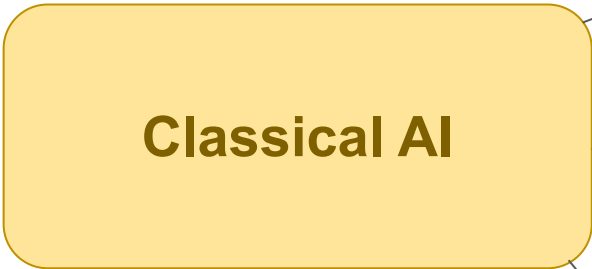


Symbolic Representation of Output



Natural Language Answer

Execute



Classical AI

LLMs as Translators !



Sat LM

Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

Translation



SatLM

LLM (Parse)



```
total_pokemon = 339
stan_pokemon = Variable()
alex_pokemon = stan_pokemon + 5
stan_pokemon = adelwolfe_pokemon * 4 - 13
total_pokemon = alex_pokemon + stan_pokemon +
adelwolfe_pokemon
result = stan_pokemon
solve(result)
```

Declarative Specification



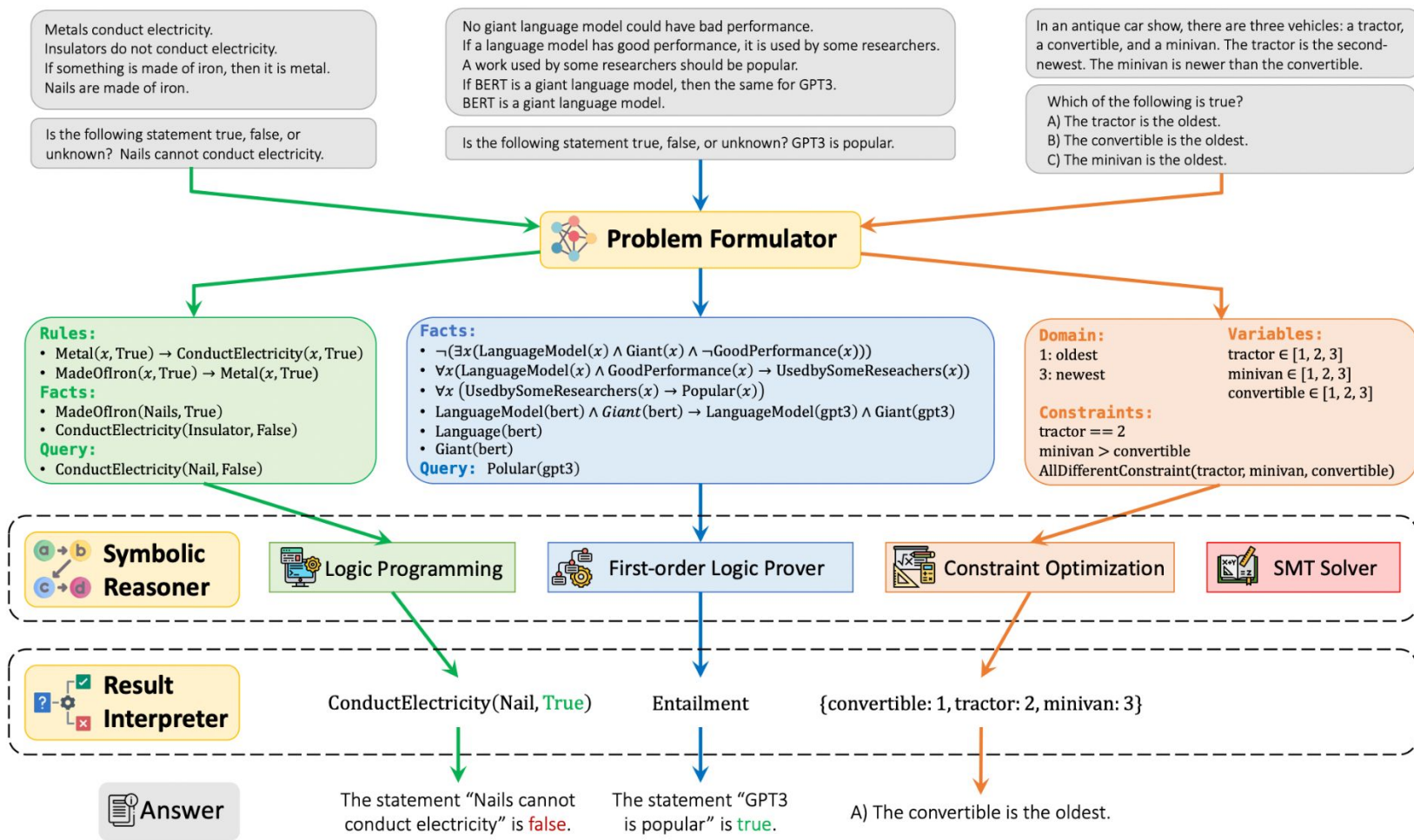
FOL Formulas

Solver (Plan + Execute)

Z3 result = 147 ✓

Robust & Faithful Reasoning





[LOGIC-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning. Pan et al., 2023](#)

Where are we ?

Is a model with 100 Billion parameters not useful for anything else



Natural Language Task



LLM



LLMs are mere translators !

Reasoning Task

LLM



PAL

Logic-LM

Sat-LM

Improve their Reasoning



Let the LLM play to its strengths and delegate remaining tasks !

CoT

SC

ToT

LLMs as Idea Generators !

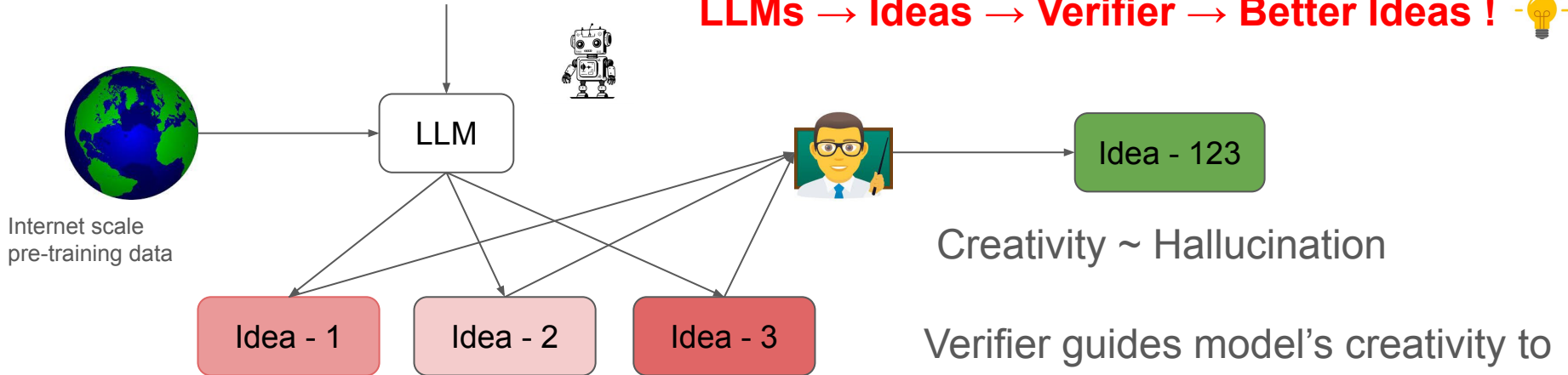
Reasoning requires being creative !

“The way to get good ideas is to get lots of ideas, and throw the bad ones away.”

- Linus Pauling



Coding Assignment
building a neural network for question
table answering



LLMs → Ideas → Verifier → Better Ideas ! 💡

Creativity ~ Hallucination

Verifier guides model's creativity to ensure correctness

Bin Packing

- Given a set of blocks of different sizes and bins, decide which bin to place the block in
- Minimize the total number of bins used
- Practical Eg: Job Scheduling on clusters
- Combinatorial Optimization: NP Hard
- Online Setting → Heuristics



FunSearch Set up: Online Bin Packing as a Program

```
def main(problem):  
    """Runs `solve` on online 1d bin packing instance,  
    ↪ and evaluates the output."""  
    bins = problem.bins  
    # Packs `problem.items` into `bins` online.  
    for item in problem.items:  
        # Extract bins that have space to fit item.  
        valid_bin_indices =  
            ↪ utils_packing.get_valid_bin_indices(item,  
            ↪ bins)  
        best_index = solve(item,  
            ↪ bins[valid_bin_indices])  
        # Add item to the selected bin.  
        bins[valid_bin_indices[best_index]] -= item  
    return evaluate(bins, problem)
```

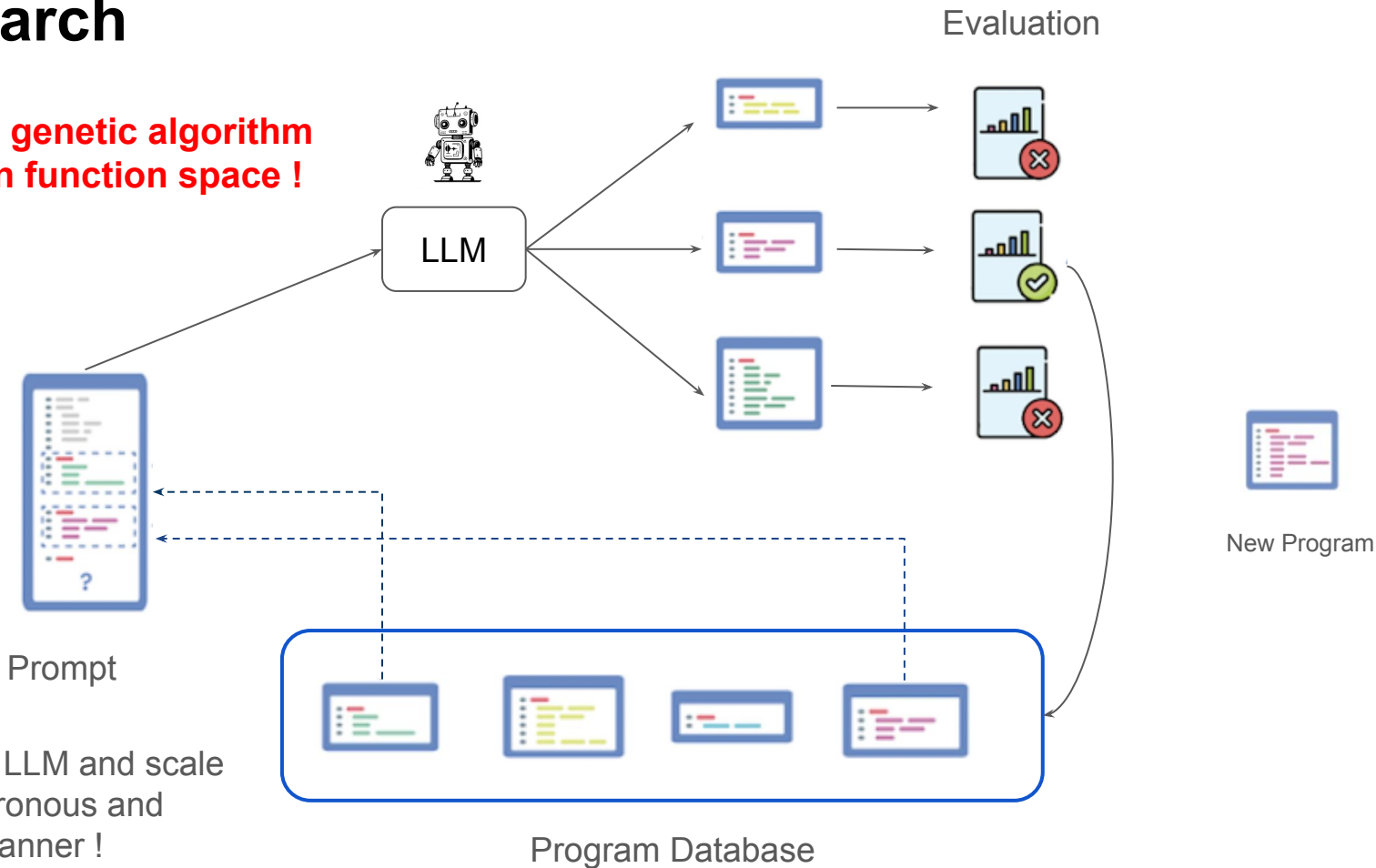
```
def heuristic(item, bins):  
    """Returns priority with which we want to add  
    ↪ `item` to each bin."""  
    return -(bins - item)
```

Use LLMs to discover new heuristics
and evaluate these !

```
def evaluate(bins, problem):  
    """Returns the negative of the number of bins  
    ↪ required to pack items in `problem`."""  
    if utils_packing.is_valid_packing(bins, problem):  
        return -utils_packing.count_used_bins(bins,  
            ↪ problem)  
    else:  
        return None
```

FunSearch

LLM guided genetic algorithm
for search in function space !



Use a cheap LLM and scale
in an asynchronous and
distributed manner !

FunSearch Results

```
def heuristic(item: float, bins: np.ndarray) -> np.ndarray:
    """Online bin packing heuristic discovered with FunSearch."""
    score = 1000 * np.ones(bins.shape)
    # Penalize bins with large capacities.
    score -= bins * (bins-item)
    # Extract index of bin with best fit.
    index = np.argmin(bins)
    # Scale score of best fit bin by item size.
    score[index] *= item
    # Penalize best fit bin if fit is not tight.
    score[index] -= (bins[index] - item)**4
    return score
```



Best-fit heuristic



FunSearch

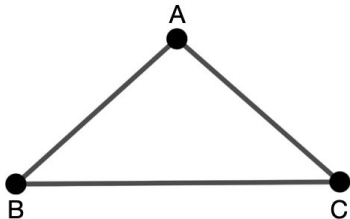
FunSearch Results

“

The solutions generated by FunSearch are far conceptually richer than a mere list of numbers. When I study them, I learn something.

JORDAN ELLENBERG, COLLABORATOR AND PROFESSOR OF MATHEMATICS AT THE UNIVERSITY OF WISCONSIN-MADISON

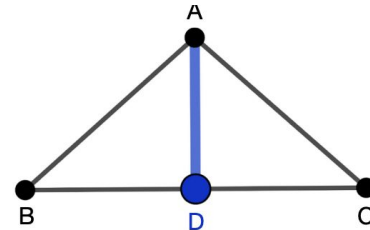
Geometry: Classical AI approach



“Let ABC be any triangle with $AB = AC$.
Prove that $\angle ABC = \angle BCA$.”

Construction lead to
large branching factors

LLMs can suggest constructions to guide search 



► Construct D: midpoint BC,
 $AB=AC, BD = DC, AD=AD \Rightarrow \angle ABD=\angle DCA$ [1]
 [1], B C D collinear $\Rightarrow \angle ABC=\angle BCA$

Construction	Description
$X = \text{angle bisector}(A, B, C)$	Construct a point X on the angle bisector of $\angle ABC$
$X = \text{angle mirror}(A, B, C)$	Construct a point X such that BC is the bisector of $\angle ABX$
$X = \text{circle}(A, B, C)$	Construct point X as the circumcenter of A, B, C
$A, B, C, D = \text{eq_quadrilateral}()$	Construct quadrilateral ABCD with $AD = BC$
$A, B, C, D = \text{eq_trapezoid}()$	Construct trapezoid ABCD with $AD = BC$
$X = \text{eqtriangle}(B, C)$	Construct X such that XBC is an equilateral triangle

Perform Search

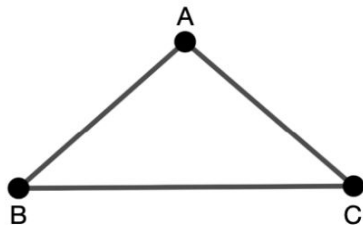
- $\text{cyclic}(E, A, D, H) \rightarrow \angle EAH = \angle EDH$
- $X = \text{eqtriangle}(B, C) \rightarrow \angle XBC = 60$

....

Represent problems
Symbolically

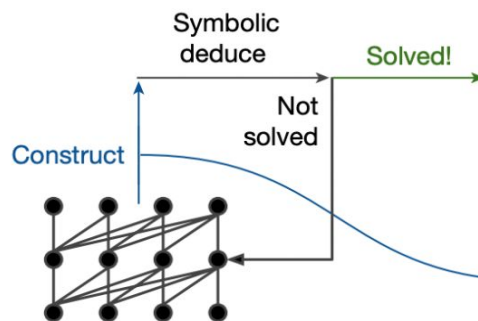
AlphaGeometry

a A simple problem



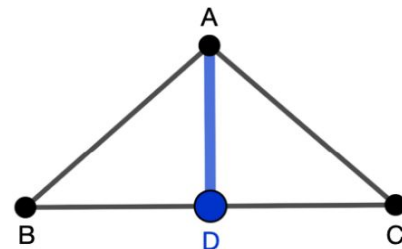
“Let ABC be any triangle with $AB = AC$.
Prove that $\angle ABC = \angle BCA$.”

b AlphaGeometry



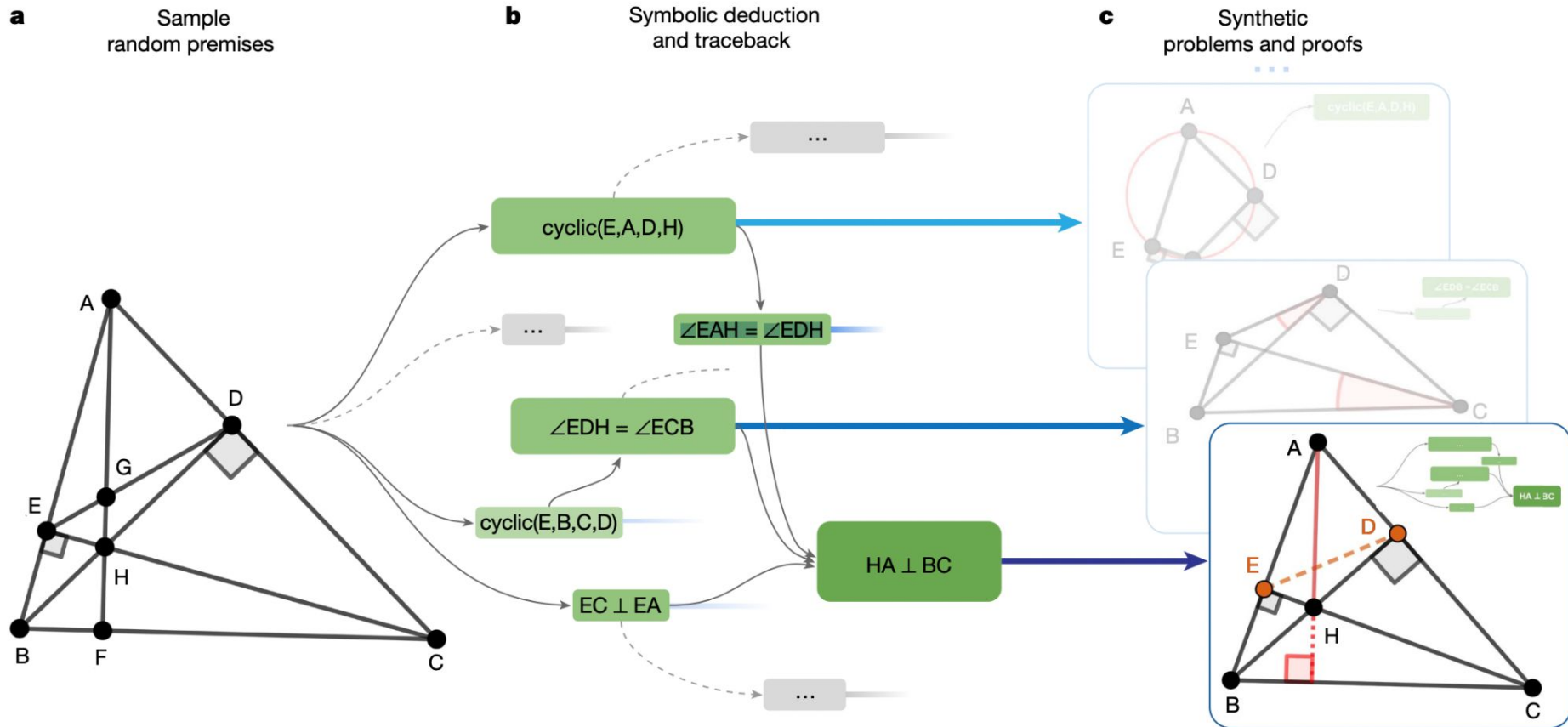
c Language model

d Solution

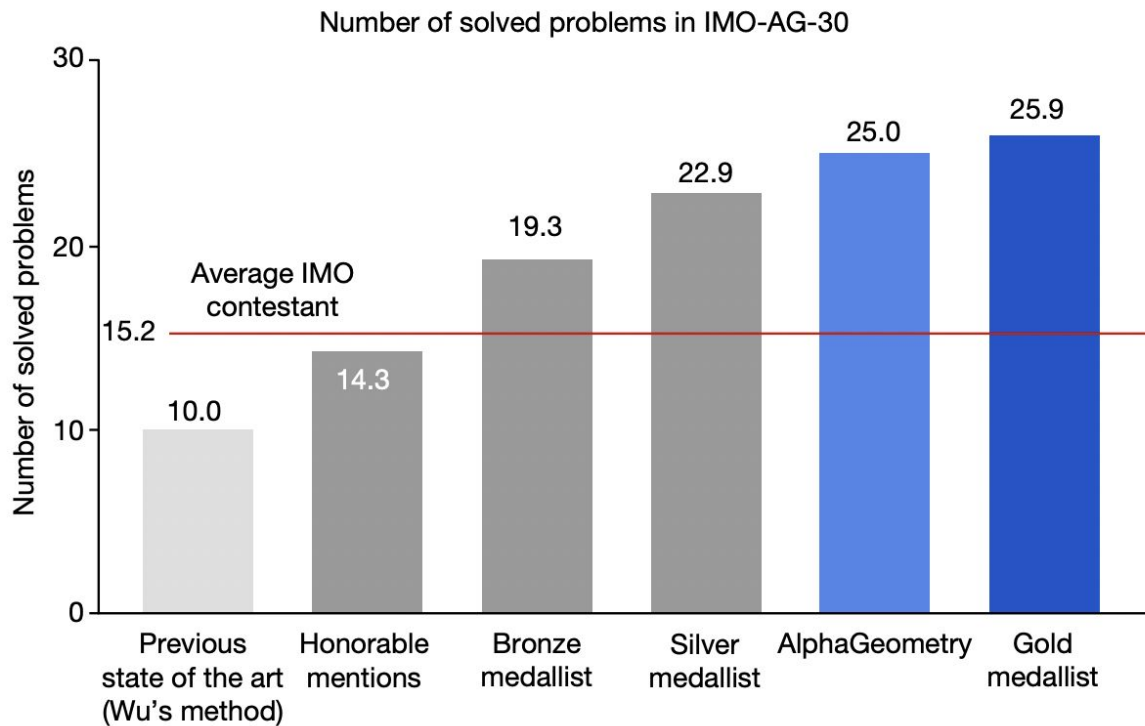


Construct D: midpoint BC,
 $AB=AC, BD = DC, AD=AD \Rightarrow \angle ABD=\angle DCA$ [1]
[1], **B C D collinear** $\Rightarrow \angle ABC=\angle BCA$

Generating Synthetic Data



AlphaGeometry Results



Conclusion

- Reasoning → What ? Why ?
- Why LLMs struggle at reasoning ?
- Improving Reasoning
 - Prompting
 - CoT
 - ToT
 - Decoding
 - SC
 - Tools
 - PAL → LLMs can write code
 - Logic-LM → LLMs as translators
 - FunSearch, AlphaGeometry → LLMs as idea generators