



CS 839: Foundation Models **Prompting**

Fred Sala

University of Wisconsin-Madison

Sept. 25, 2025

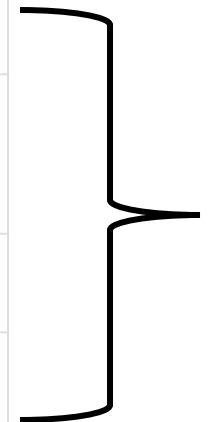
Announcements

- **Logistics:**

- Homework 1 is ongoing!
 - Due **Oct. 2** end of day.
- Upcoming: presentation information

- **Class roadmap:**

Thursday Sept. 25	Prompting
Tuesday Sept. 30	Specialization
Thursday Oct. 2	Alignment
Tuesday Oct. 7	RLVR
Thursday Oct. 9	Efficient Training



Language & Foundation Models

Outline

- **Finish Improving and Extending Prompting**

- Searching for good prompts, techniques for continuous/soft prompts, ensembling

- **Intro to Chain-of-Thought**

- Basic idea, zero-shot and few-shot, choosing examples for few-shot, tree-of-thoughts

- **Ingredients For CoT + External Improvements**

- Scale, manual vs. auto-rationales, rationale correctness. Programs, tools, etc.

Outline

- **Finish Improving and Extending Prompting**

- Searching for good prompts, techniques for continuous/soft prompts, ensembling

- **Intro to Chain-of-Thought**

- Basic idea, zero-shot and few-shot, choosing examples for few-shot, tree-of-thoughts

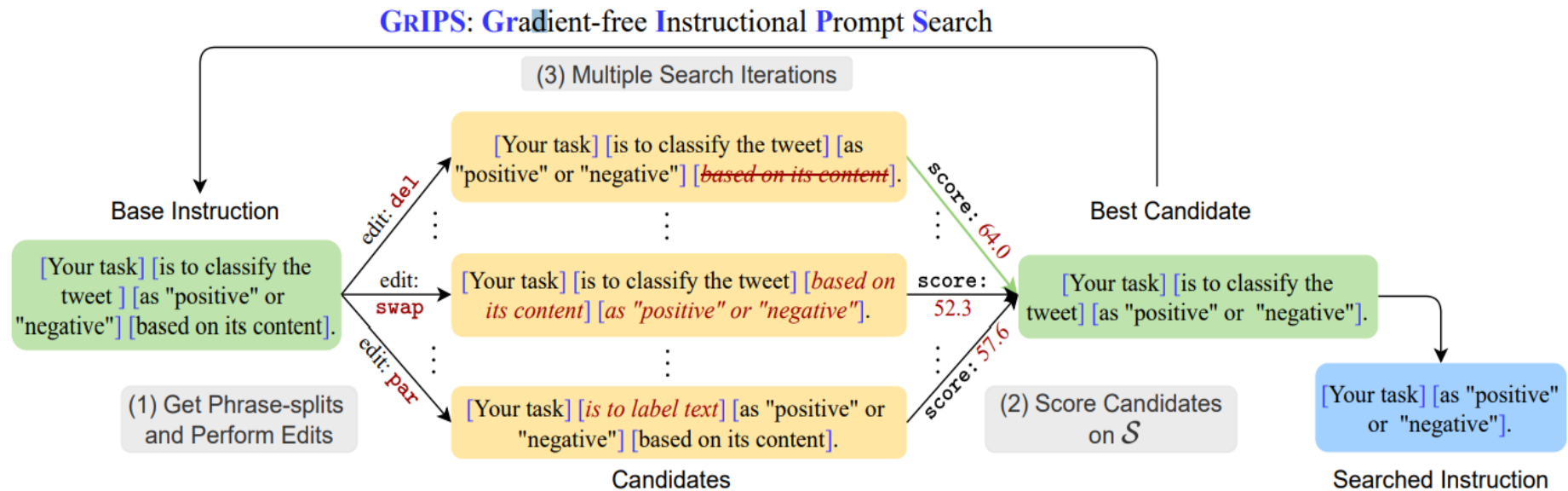
- **Ingredients For CoT + External Improvements**

- Scale, manual vs. auto-rationales, rationale correctness. Programs, tools, etc.

Hard Prompting: Discrete Optimization

Sometimes, can avoid gradients

- Random search
- Greedy



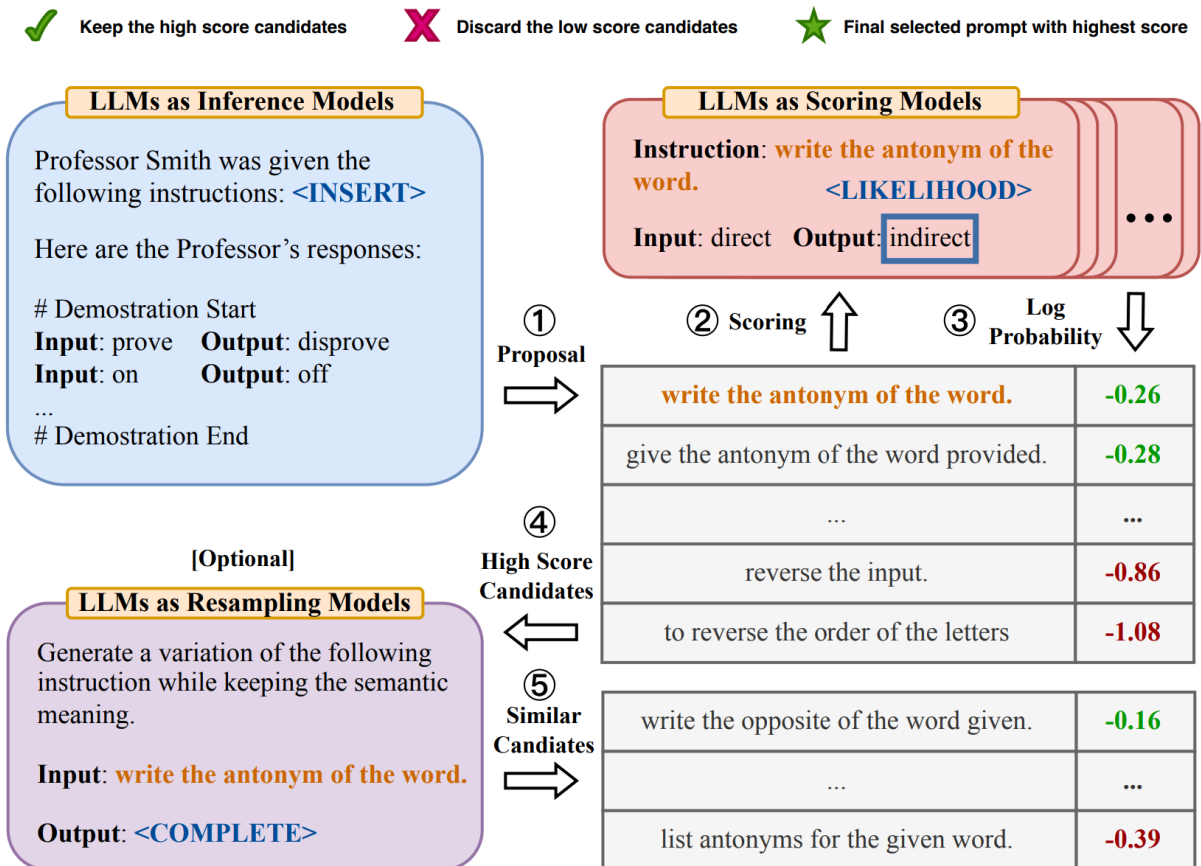
Auto-Prompting

LLMs as “prompt engineers” (Zhou et al, ‘23)

- Use an LLM to generate candidate instructions (prompts)
- Evaluate them externally
- Select best candidate.
- Optionally iterate.

Example Output:

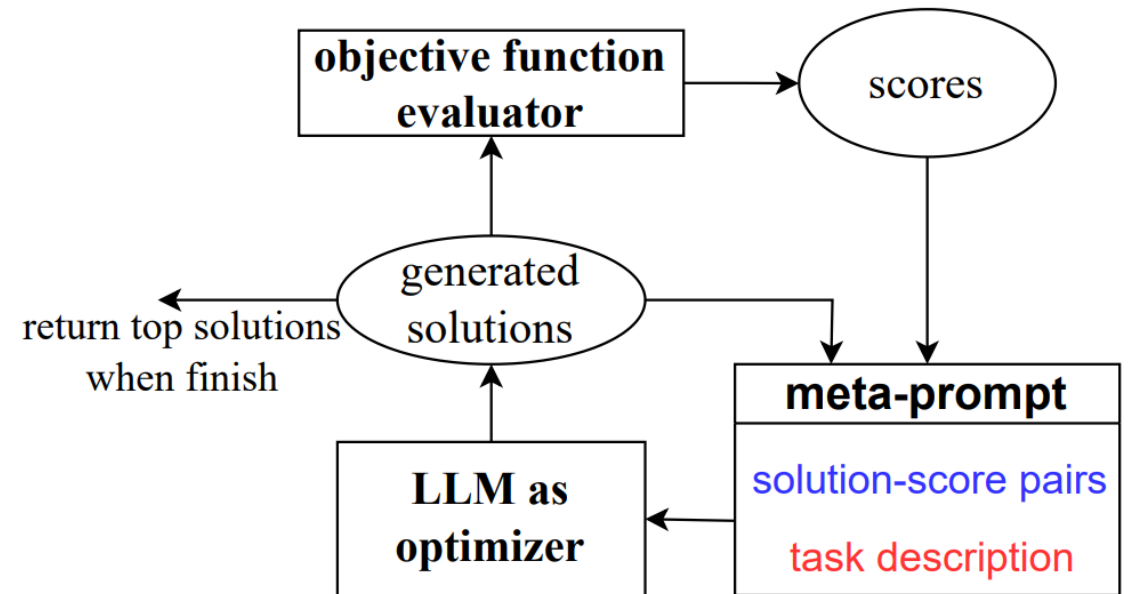
- “Let’s work this out in a step by step way to be sure we have the right answer.”



Auto-Prompting

LLMs as “optimizers” (Yang et al, ‘23)

- Use the LLM to guess solutions to an optimization problem
 - Evaluate them externally
 - Run in a loop with few-shot
-
- Can use for
 - Standard optimization problems
 - Tougher settings like prompts



LLMs as Optimizers: Example

Example:

I have some texts along with their corresponding scores. The texts are arranged in ascending order based on their scores, where higher scores indicate better quality.

text:
Let's figure it out!
score:
61

text:
Let's solve the problem.
score:
63

(... more instructions and scores ...)

The following exemplars show how to apply your text: you replace <INS> in each input with your text, then read the input and give an output. We say your output is wrong if your output is different from the given output, and we say your output is correct if they are the same.

input:
Q: Alannah, Beatrix, and Queen are preparing for the new school year and have been given books by their parents. Alannah has 20 more books than Beatrix. Queen has $\frac{1}{5}$ times more books than Alannah. If Beatrix has 30 books, how many books do the three have together?
A: <INS>
output:
140

(... more exemplars ...)

Write your new text that is different from the old ones and has a score as high as possible. Write the text in square brackets.



Meta-instructions



Trajectory points



Problem to be solved

LLMs as Optimizers: Prompt Optimization

Resulting trajectory

- “Solve the following problems using the given information.” at Step 2 with training accuracy 59.8;
- “Solve the following problems by applying the given information and using the appropriate mathematical operations.” at Step 3 with training accuracy 64.0;
- “Let’s read the problem carefully and identify the given information. Then, we can create an equation and solve for the unknown variable.” at Step 4 with training accuracy 67.0;
- “I’m always down for solving a math word problem together. Just give me a moment to read and understand the problem. Then, I’ll create an equation that models the problem, which I’ll solve for the unknown variable. I also may or may not use some helpful diagrams or visuals to understand the problem. Lastly, be sure to allow me some time to carefully check my work before submitting any responses!” at Step 29 with training accuracy 70.1.

Ours

PaLM 2-L

PaLM
2-T-TT

A_begin

Take a deep breath and work on this problem step-by-step.

80.2

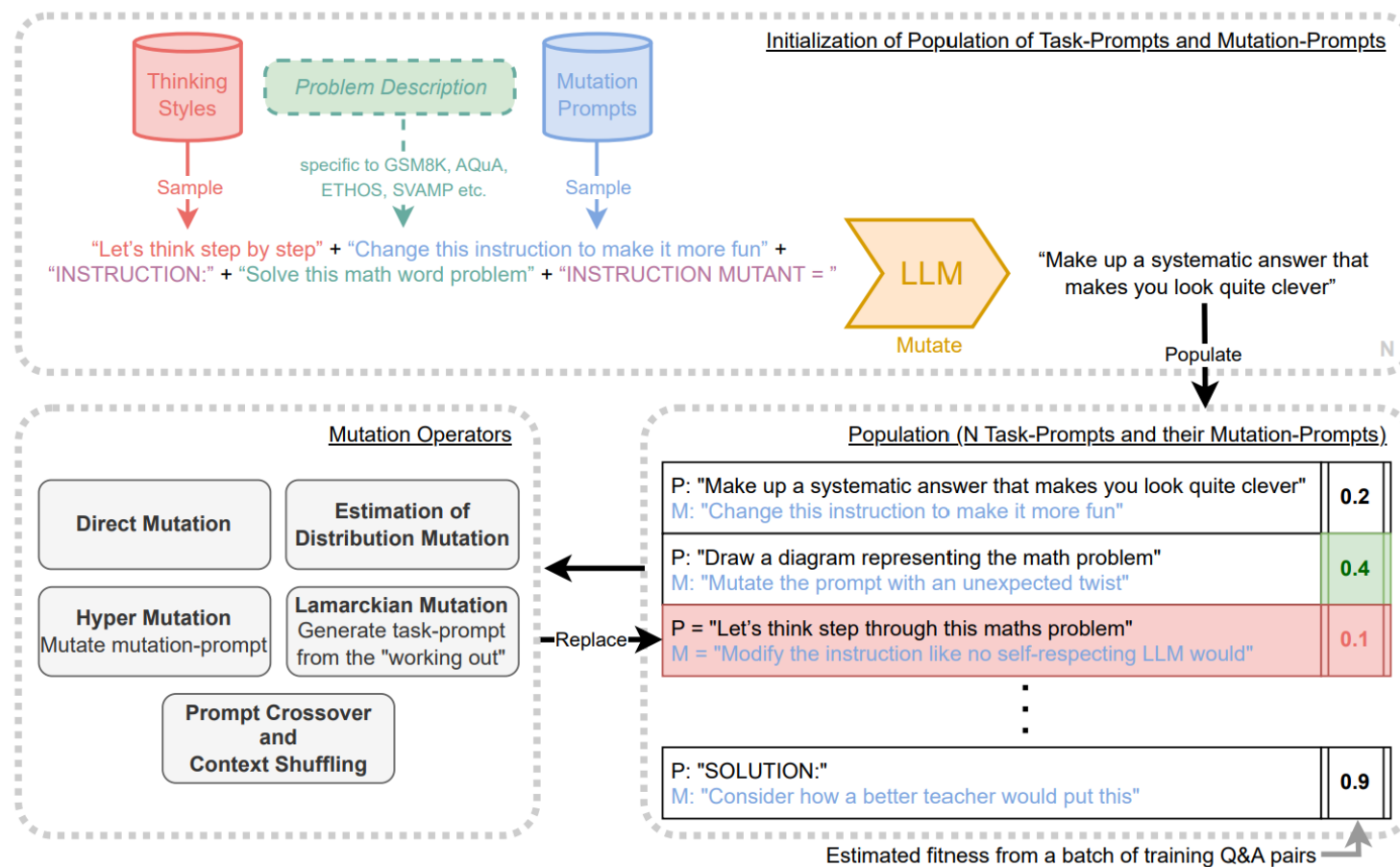
More Auto-Prompting Work

Recall: search for hard prompts: tough optimization problem

- Lots of classic search methods only require notion of “**neighbors**” and **evaluation** function access
 - Hill-climbing
 - Simulated annealing
 - Genetic algorithms
- “**Promptbreeder**”: an approach via genetic algorithms
- Show all your working. II. You should use the correct mathematical notation and vocabulary, where appropriate. III. You should write your answer in full sentences and in words. IV. You should use examples to illustrate your points and prove your answers. V. Your workings out should be neat and legible

More Auto-Prompting Work

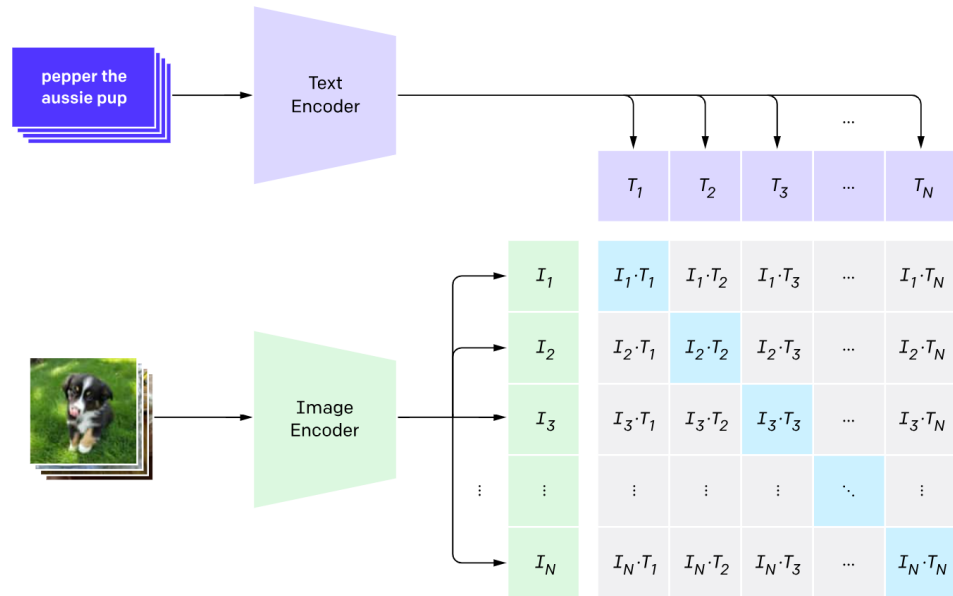
“Promptbreeder”: an approach via genetic algorithms



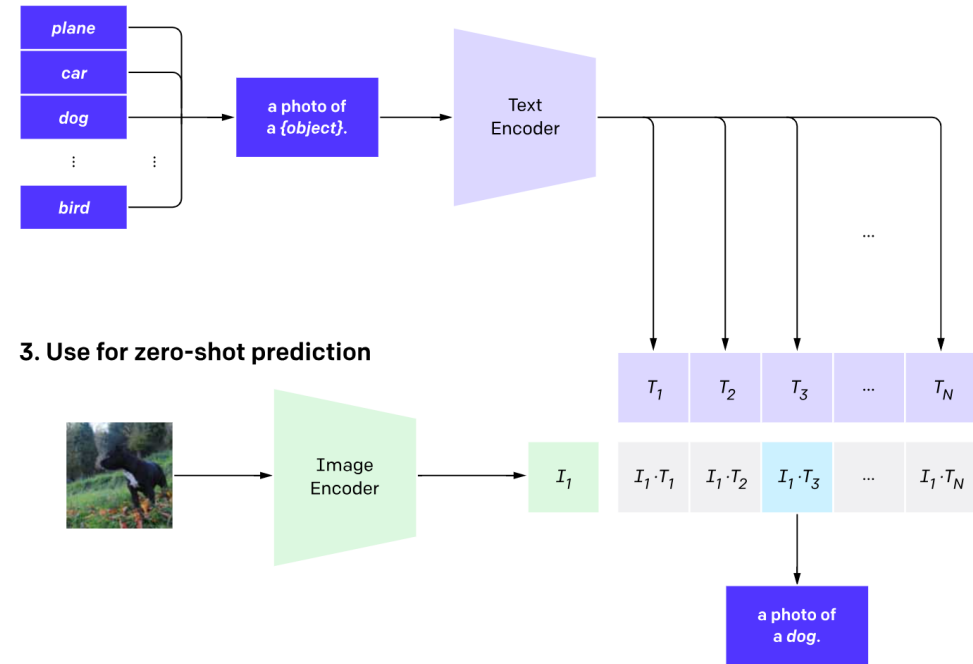
Prompting VLMs and Multimodal Models

Training and prediction in CLIP-style VLMs

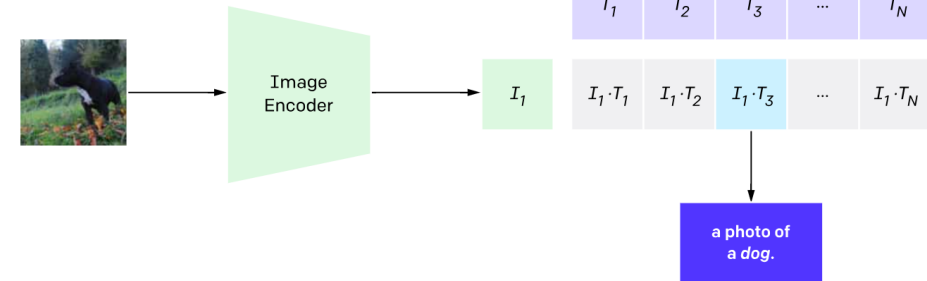
1. Contrastive pre-training



2. Create dataset classifier from label text



3. Use for zero-shot prediction



How to Prompt VLMs?

Standard way: use pre-defined templates

- E.g., “a photo of a [X]”
- But, might struggle...

SUN397

television studio (90.2%) Ranked 1 out of 397 labels



✓ a photo of a **television studio**.

✗ a photo of a **podium indoor**.

✗ a photo of a **conference room**.

✗ a photo of a **lecture room**.

✗ a photo of a **control room**.

LLMs to Improve VLMs: Description

Static class descriptions may fail...

- Replace with descriptive features (Menon and Vondrick, '23)
 - Instead of “tiger”, include “stripes, claws, ...”

School bus

- a large, yellow vehicle
- the words "school bus" written on the side
- a stop sign that deploys from the side of the bus
- flashing lights on the top of the bus
- large windows

Shoe store

- a building with a sign that says "shoe store"
- a large selection of shoes in the window
- shoes on display racks inside the store
- a cash register
- a salesperson or customer

Volcano

- a large, cone-shaped mountain
- a crater at the top of the mountain
- lava or ash flowing from the crater
- a plume of smoke or ash rising from the crater

Barber shop

- a building with a large, open storefront
- a barber pole or sign outside the shop
- barber chairs inside the shop
- mirrors on the walls
- shelves or cabinets for storing supplies
- a cash register
- a waiting area for customers

Cheeseburger

- a burger patty
- cheese
- a bun
- lettuce
- tomato
- onion
- pickles
- ketchup
- mustard

Violin

- a stringed instrument
- typically has four strings
- a wooden body
- a neck and fingerboard
- tuning pegs
- a bridge
- a soundpost
- f-holes
- a bow

Pirate ship

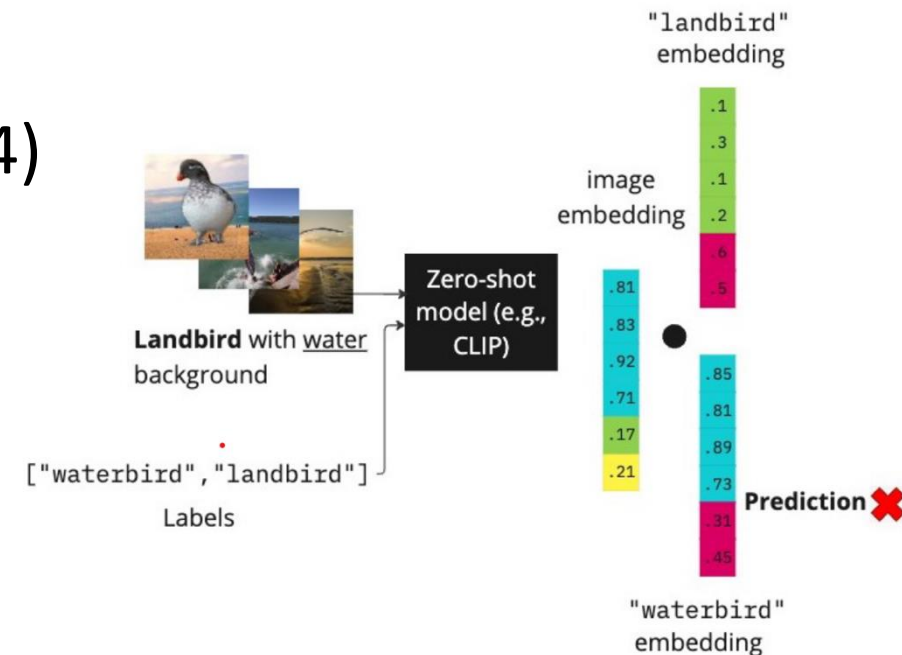
- a large, sailing vessel
- a flag with a skull and crossbones
- cannons on the deck
- a wooden hull
- portholes
- rigging
- a crow's nest

Figure 3: Examples of descriptor schema produced by GPT-3.

LLMs to Improve VLMs: Spurious Features

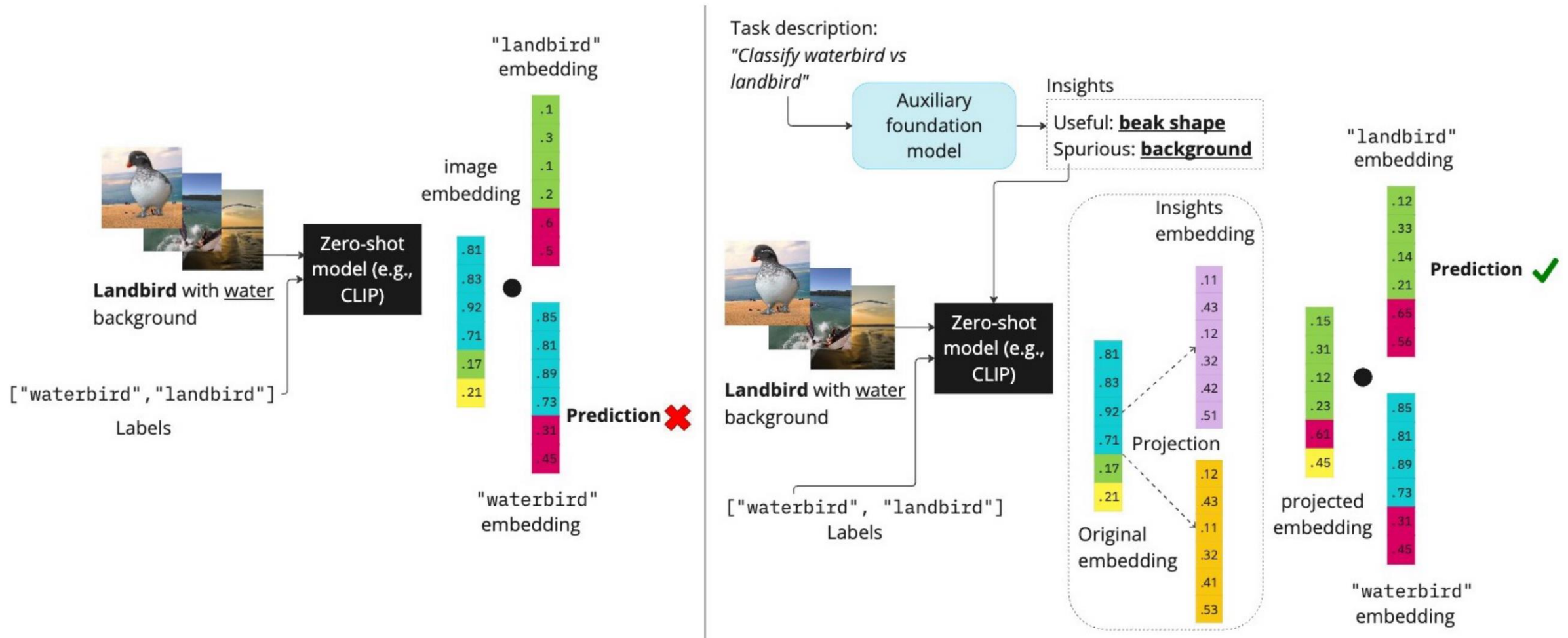
This helped with positives.

- What about **negatives** (i.e., spurious features?)
 - Example: waterbirds with CLIP
- Spurious correlations: generally a problem with all pretrained models
 - But LLMs can also tell us about this (Adila '24)



LLMs to Improve VLMs: Spurious Features

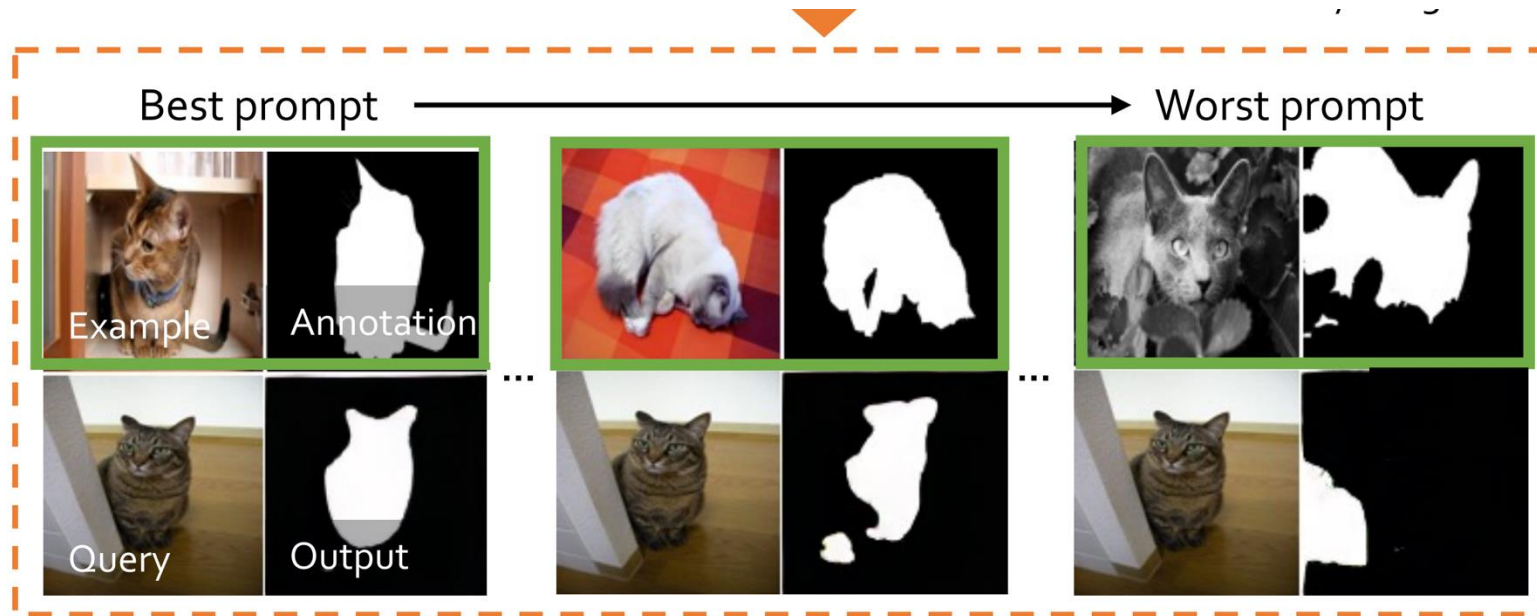
Modify embeddings used for prediction:



In-Context Learning for VLMs

Surprisingly in-context/few-shot also applies to visual models

- Standard questions apply here too:
 - How to select examples
 - What makes for a good example?



(a) Visual in-context learning is sensitive to prompt selection

Zhang et al '23



Break & Questions

Outline

- **Finish Improving and Extending Prompting**

- Searching for good prompts, techniques for continuous/soft prompts, ensembling

- **Intro to Chain-of-Thought**

- Basic idea, zero-shot and few-shot, choosing examples for few-shot, tree-of-thoughts

- **Ingredients For CoT + External Improvements**

- Scale, manual vs. auto-rationales, rationale correctness. Programs, tools, etc.

Chain-of-Thought

Performing complex reasoning is hard. Help the model:

Standard Prompting

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

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. ✅

Chain-of-Thought: Applications

Some things it can be used for:

- **Math problems**
- **Commonsense reasoning**
- **Symbolic reasoning**

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).

CSQA (commonsense)

Q: Sammy wanted to go to where the people were. Where might he go?
Options: (a) race track (b) populated areas (c) desert (d) apartment (e) roadblock

A: The answer must be a place with a lot of people. Race tracks, desert, apartments, and roadblocks don't have a lot of people, but populated areas do. So the answer is (b).

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.

Date Understanding

Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.

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.

SayCan (Instructing a robot)

Human: How would you bring me something that isn't a fruit?

Explanation: the user wants something to eat that isn't a fruit. An energy bar is not a fruit, so I will bring the user an energy bar.

Plan: 1. find(energy bar) 2. pick(energy bar) 3. find(user) 4. put(energy bar) 5. done().

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.

Chain-of-Thought: Zero-Shot

No examples shown; encourage model to decompose

- Add to prompt: “Let’s think step by step” before each answer
- For answer extraction, add prompts like “Therefore, the answer (arabic numerals) is” (Kojima et al ‘23)

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The answer is 8. ✗

(b) Few-shot-CoT

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is

(Output) 8 ✗

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

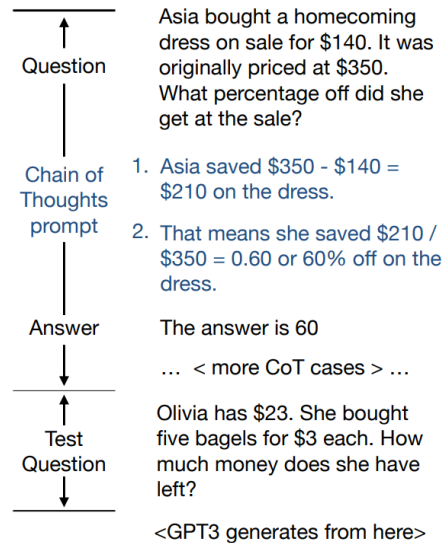
A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

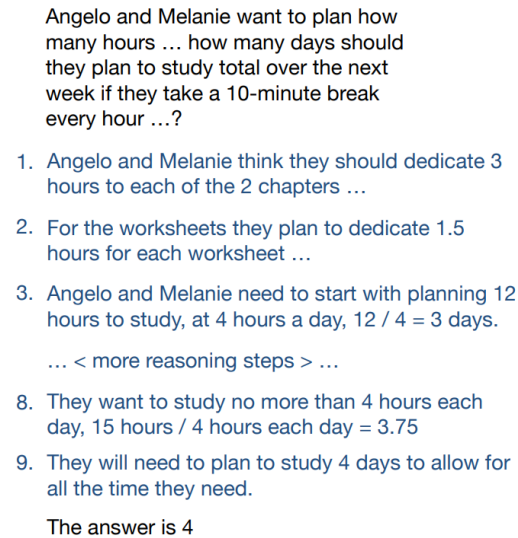
Chain-of-Thought: Few-Shot Examples

As before, we must choose few-shot examples.

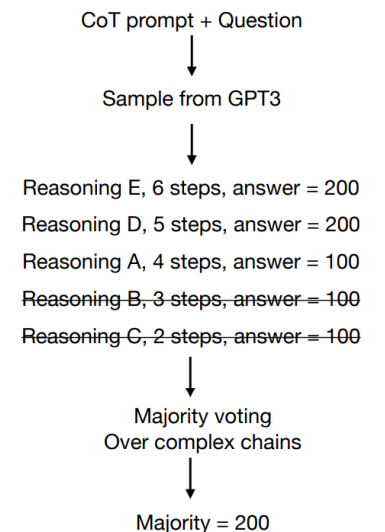
- More structured than simple semantic similarity
- *Complexity-based* prompting.
 - “[S]imply choose complex prompts over simple ones.”
- Prompting: include most steps. Ensembling: MV over set of most complex chains.



A. Workflow of chain of thoughts prompting



B. Example complex chain, 9 reasoning steps

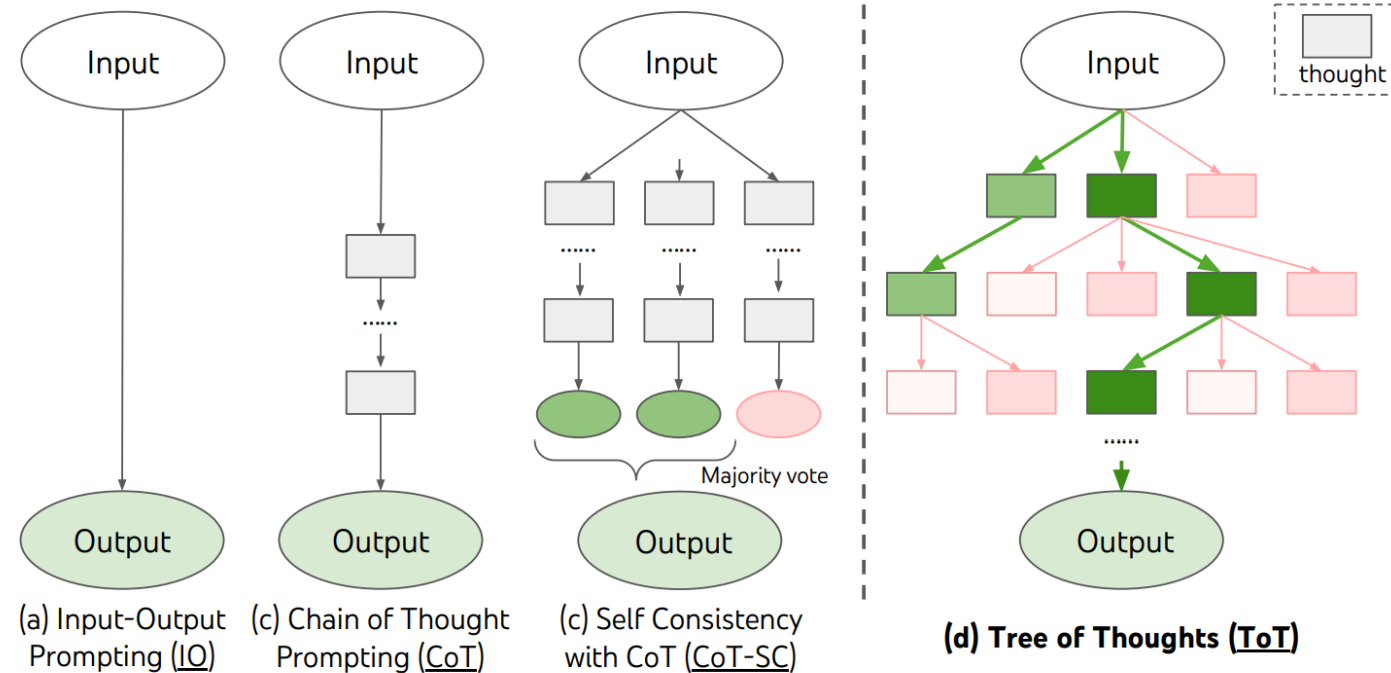


C. Complexity-based consistency

Chain-of-Thought: Generalizations

How do we really “reason”?

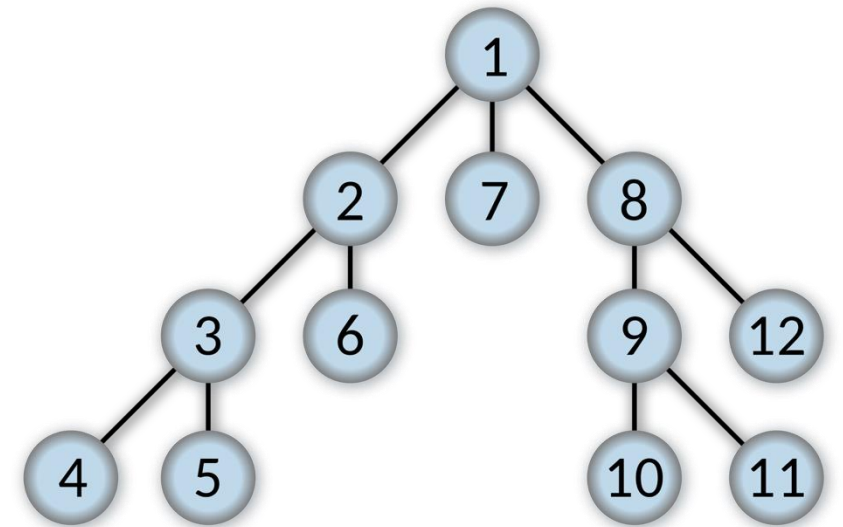
- Not really by sampling a bunch of chains...



Chain-of-Thought: Generalizations

Tree-of-thoughts **basic idea**:

- **Notation:** thoughts z_1, z_2, \dots, z_n bridge x and y
- Comparison to other methods:
 - Vanilla CoT: sample $z_i \sim p_\theta(z_i \mid x, z_1, \dots, z_{i-1})$, $y \sim p_\theta(y \mid x, z_1, \dots, z_n)$
 - CoT Self-Consistency: sample multiple times, take majority vote
- Idea: create a state $s=[x, z_1, \dots, z_n]$
- Generate multiple candidates for next state
 - Then run standard search (i.e., BFS, DFS, A*)

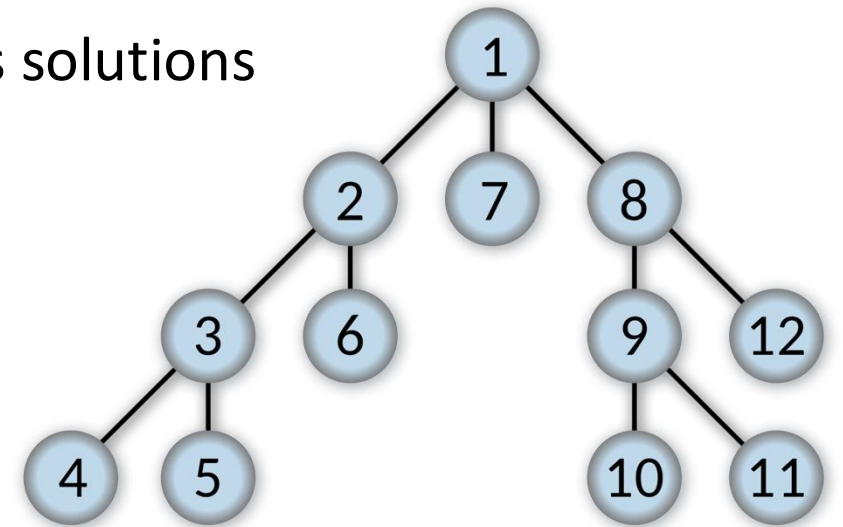


Drichel (Wiki)

Chain-of-Thought: **Generalizations**

Tree-of-thoughts **key aspects**:

- **Thought decomposition**: how big zs should be
- **Thought generation**: obtaining the next sample
 - Try to avoid duplication
- **State evaluation**: How close are we to solution?
 - Recall heuristics for search from CS 540
 - Either use LM itself, or vote/weighted vote across solutions
- **Search**: BFS or DFS
 - Or more advanced search methods



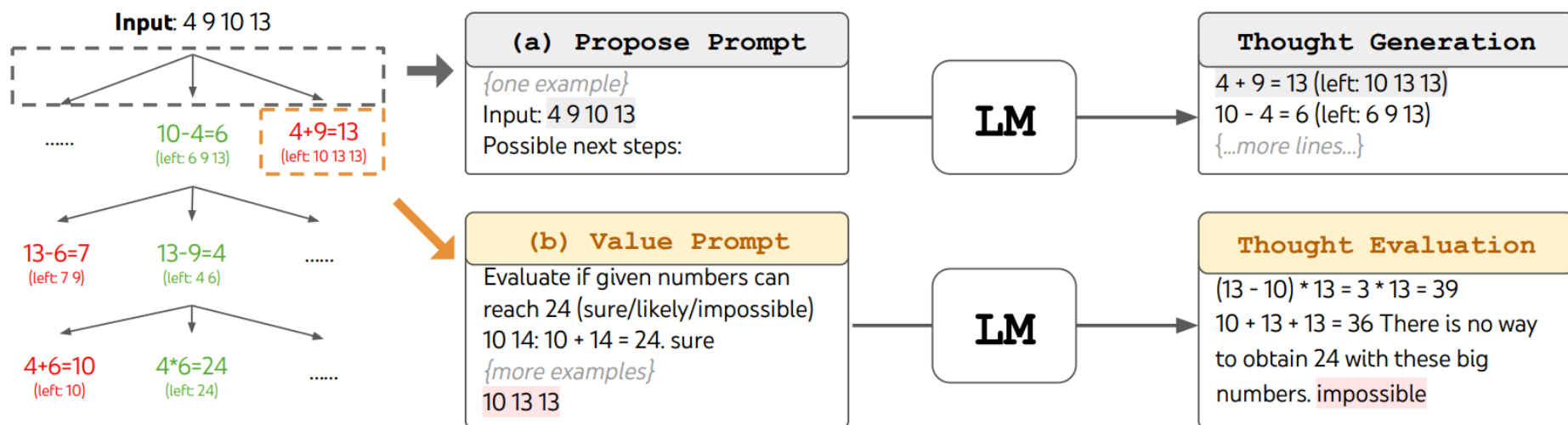
Drichel (Wiki)

Chain-of-Thought: Generalizations

Tree-of-thoughts example:

4.1 Game of 24

Game of 24 is a mathematical reasoning challenge, where the goal is to use 4 numbers and basic arithmetic operations (+-*/) to obtain 24. For example, given input “4 9 10 13”, a solution output could be “(10 - 4) * (13 - 9) = 24”.





Break & Questions

Outline

- **Finish Improving and Extending Prompting**

- Searching for good prompts, techniques for continuous/soft prompts, ensembling

- **Intro to Chain-of-Thought**

- Basic idea, zero-shot and few-shot, choosing examples for few-shot, tree-of-thoughts

- **Ingredients For CoT + External Improvements**

- Scale, manual vs. auto-rationales, rationale correctness. Programs, tools, etc.

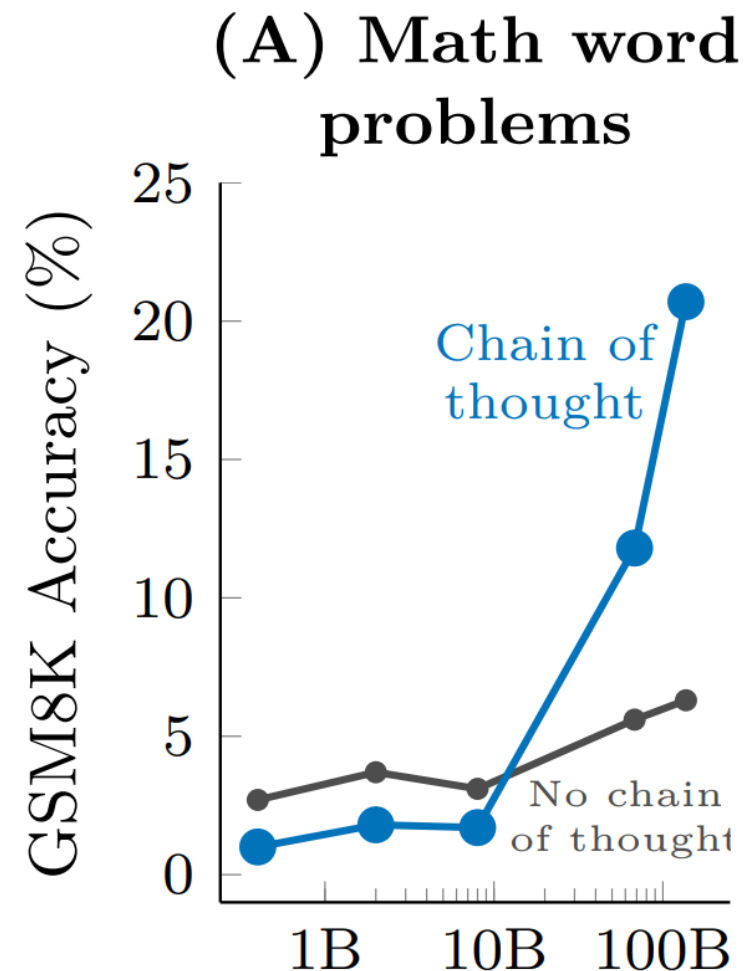
What Matters for CoT? Scale?

Do all language models exhibit CoT behavior?

A: No. Shows up only at certain sizes

- “Emergent behavior”
- But... we can use RL to speed up appearance
- Example: CoT does not help until ~10B

(Model: LaMDA, Dataset: Math)



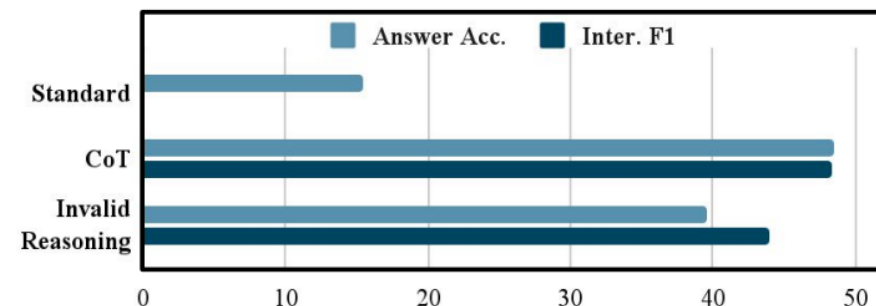
What Matters for CoT? Correctness?

Does reasoning have to work in provided examples?

A: **No**. Invalid reasoning in CoT still helps

- Versus no CoT
- What does matter
 - Relevant steps (still bridge gap)
 - Steps follow each other

	In-context Demonstration	Inference by LLM
Query	Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?	Julie is reading a 120-page book. Yesterday, she was able to read 12 pages and today, she read twice as many pages as yesterday. If she wants to read half of the remaining pages tomorrow, how many pages should she read?
Standard	The answer is 39.	The answer is 18. ✗
CoT	Originally, Leah had 32 chocolates and her sister had 42. So in total they had $32 + 42 = 74$. After eating 35, they had $74 - 35 = 39$ pieces left in total. The answer is 39.	Julie is reading a 120-page book. Yesterday, she read 12 pages and today, she read 24 pages. So she read a total of $12 + 24 = 36$ pages. Now she has $120 - 36 = 84$ pages left. Since she wants to read half of the remaining pages, she should read $84 / 2 = 42$ pages. The answer is 42. ✓
Invalid Reasoning	Originally, Leah had 32 chocolates and her sister had 42. So her sister had $42 - 32 = 10$ chocolates more than Leah has. After eating 35, since $10 + 35 = 45$, they had $45 - 6 = 39$ pieces left in total. The answer is 39.	Yesterday, Julie read 12 pages. Today, she read $12 * 2 = 24$ pages. So she read a total of $12 + 24 = 36$ pages. Now she needs to read $120 - 36 = 84$ more pages. She wants to read half of the remaining pages tomorrow, so she needs to read $84 / 2 = 42$ pages tomorrow. The answer is 42. ✓

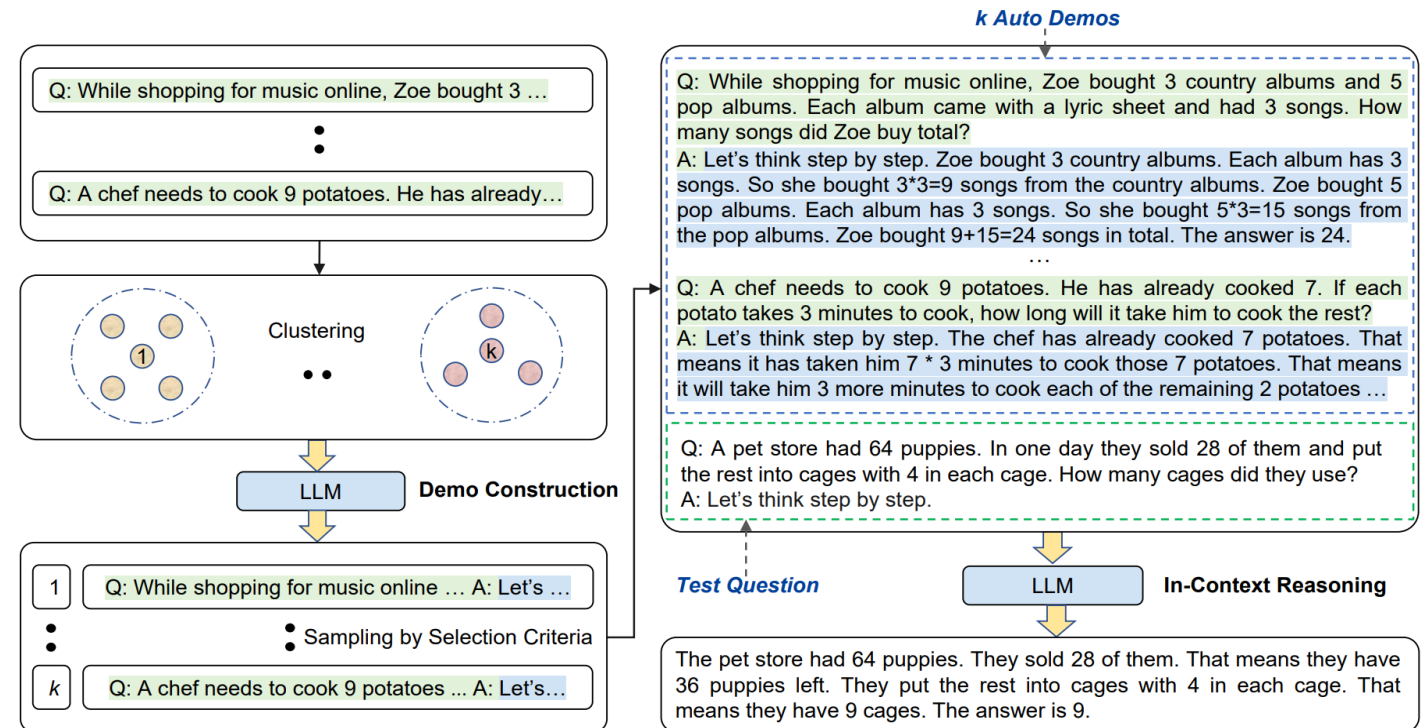


What Matters for CoT? Human signal?

Do examples have to be manually crafted?

A: **No**. Auto-CoT: generate examples to be used

- Need diversity: first cluster, then sample from each cluster



What Matters for CoT? Pretraining?

Does “reasoning” data in pretraining extend to other languages?

A: Not entirely.

	AVG	HRL	URL	EN	DE	FR	ES	RU	ZH	JA	TH	TE	BN	SW
Lang. Freq. (PaLM, %)	–	–	–	78.0	3.5	3.3	2.1	.53	.40	.38	.04	.02	.006	.005
GPT-3 (text-davinci-002)														
• DIRECT	11.7	15.1	5.7	16.0	14.8	16.8	17.2	12.4	18.0	11.2	8.8	0.8	4.4	8.8
• NATIVE-CoT	26.4	34.7	7.2	53.6	36.0	37.6	40.4	28.4	40.0	26.0	10.8	0.4	6.4	11.2
• EN-CoT	31.6	39.4	13.9	53.6	44.0	46.0	44.8	28.4	40.8	32.4	19.6	5.6	9.6	20.8
• TRANSLATE-EN	45.6	47.5	40.7	53.6	46.4	46.4	51.6	48.8	47.2	44.8	41.2	42.8	41.2	37.6
PaLM-540B														
• DIRECT	18.6	19.3	16.8	22.0	18.8	19.6	20.0	22.0	19.2	16.0	16.8	17.6	17.2	15.6
• NATIVE-CoT	48.1	47.9	44.9	62.4	49.2	46.4	56.8	48.4	46.8	40.0	52.8	45.6	46.0	35.2
• EN-CoT	51.3	52.3	46.8	62.4	53.6	51.2	58.0	55.6	46.0	49.6	49.6	46.8	46.4	44.4
• TRANSLATE-EN	55.0	56.3	51.2	62.4	57.2	55.2	60.0	59.6	55.6	50.0	50.8	49.6	53.2	51.2

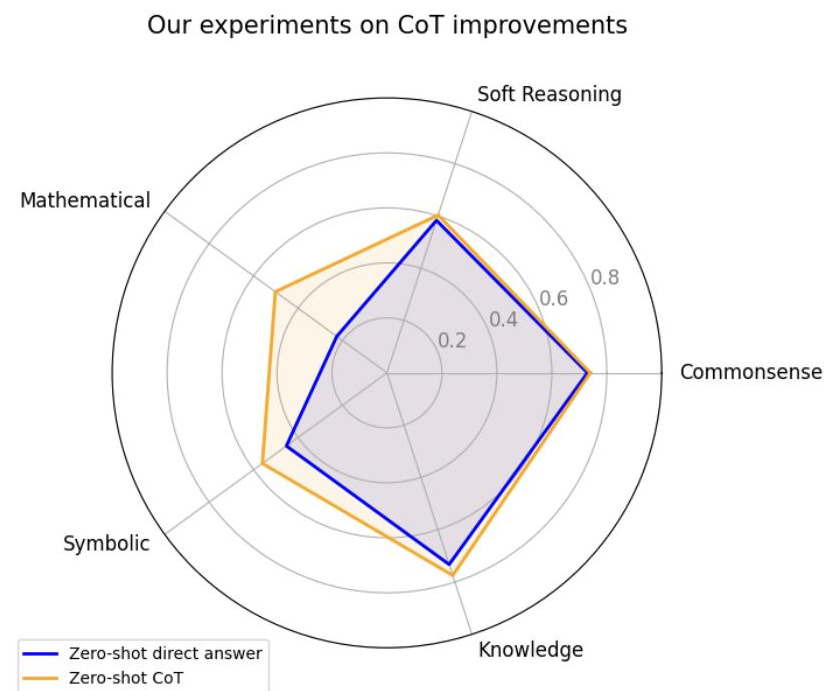
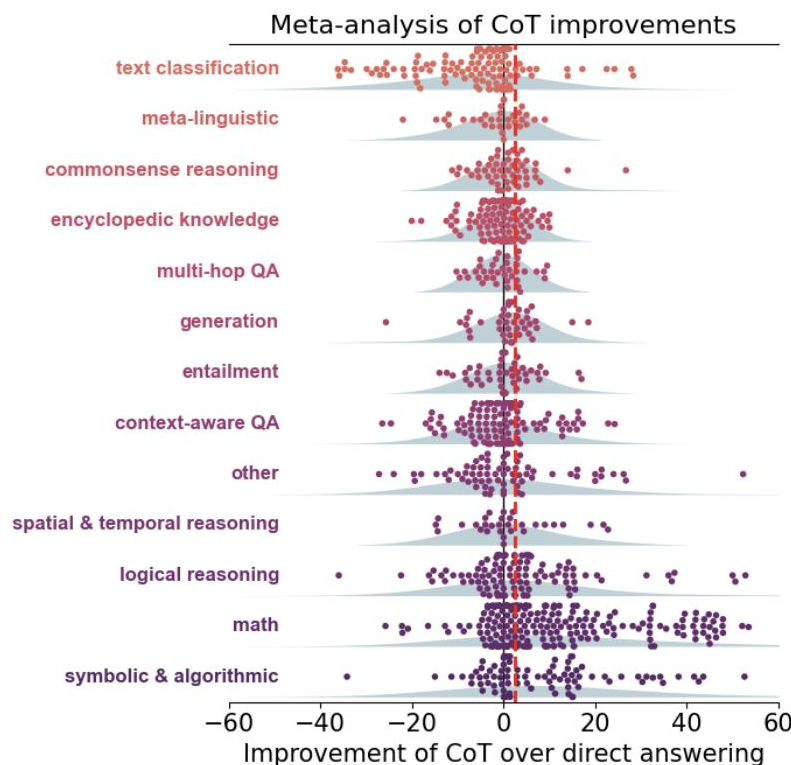
When Does CoT Actually Help?

A: Not always clear

- Mainly on math and symbolic reasoning?

To CoT or not to CoT? Chain of thought for math and symbolic reasoning

Zayne Sprague, Fangcong Yin, Juan Diego I
Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle M



Beyond the Unaided Language Models

Even when we do CoT, the language model can get things wrong.

- Often simple things... like **arithmetic**.
- How else can we help it?
- **A:** Use external tools



Tools: Program-aided LMs

Use external tools:

- Python interpreter
- How? *Interleave* the text explanations in CoT steps with lines of Python code
- LMs can already output code
 - Just need to *prompt* the right way

Chain-of-Thought (Wei et al., 2022)

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.



Gao et al '23

Program-aided Language models (this work)

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
```



Tools: Programs-aided LMs Prompts

Just need to *prompt* the right way

- Craft examples that interleave code and text

Q: Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?

```
money_initial = 23
bagels = 5
bagel_cost = 3
money_spent = bagels * bagel_cost
money_left = money_initial - money_spent
answer = money_left
```

Figure 3: Example prompt for the mathematical reasoning tasks, from the GSM8K benchmark.

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]
```

Tools: Program-of-Thoughts

Similar idea: program-of-thoughts

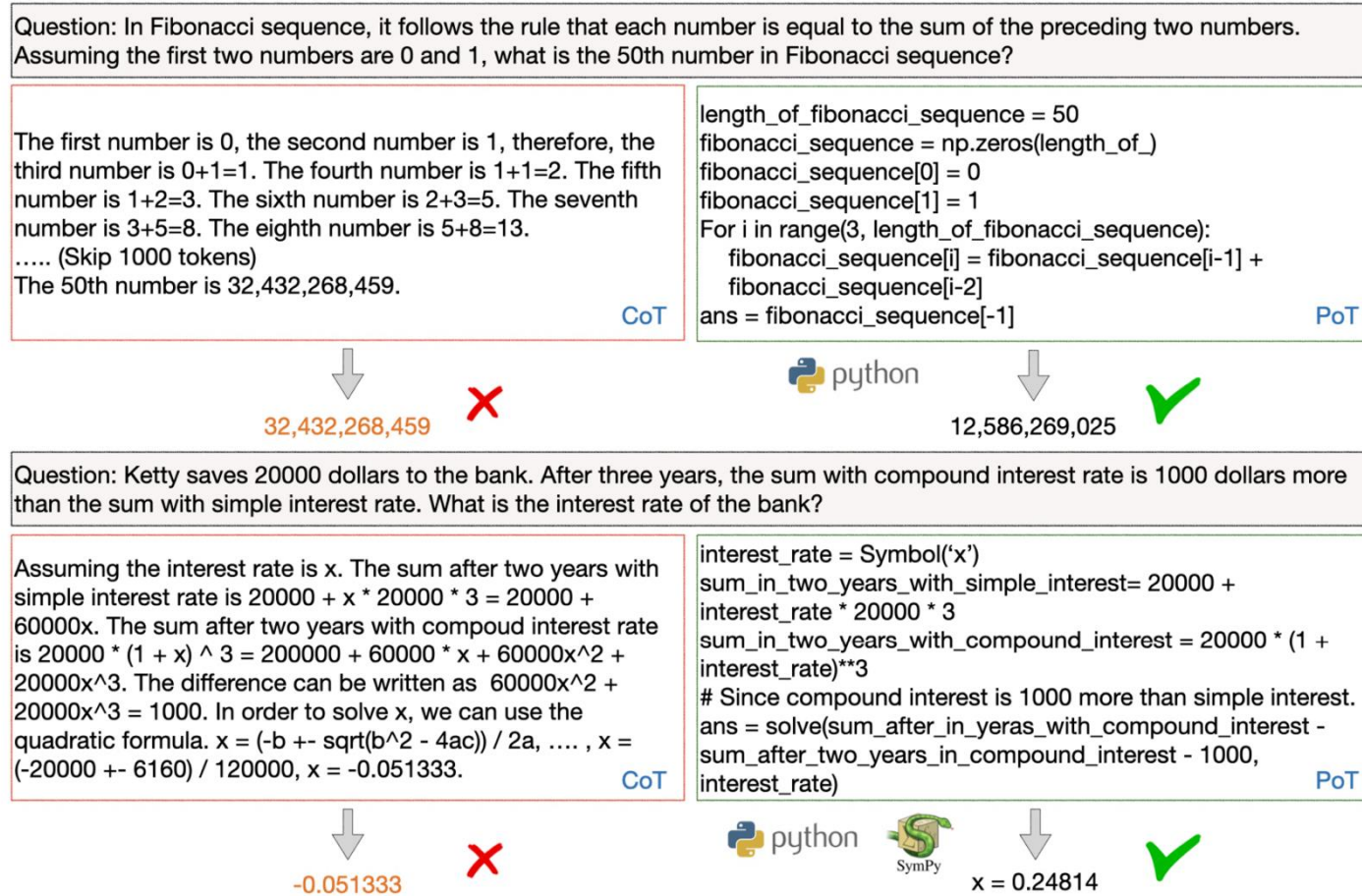


Figure 1: Comparison between Chain of Thoughts and Program of Thoughts.

Tools: More General Tools

Ideally, use more general external tools

- Without lots of human annotation
- Model should decide on its own which tool to use
- **Toolformer**: introduces API calls into the model
 - But these API calls aren't already there... so need to fine-tune
- **Model context protocol (MCP)** standardize!

Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where "question" is the question you want to ask. Here are some examples of API calls:

Input: Joe Biden was born in Scranton, Pennsylvania.

Output: Joe Biden was born in [QA("Where was Joe Biden born?")] Scranton, [QA("In which state is Scranton?")] Pennsylvania.

Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

Output: Coca-Cola, or [QA("What other name is Coca-Cola known by?")] Coke, is a carbonated soft drink manufactured by [QA("Who manufactures Coca-Cola?")] the Coca-Cola Company.

Input: x

Output:

Bibliography

- Fernando et al '23: Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, Tim Rocktäschel, "Promptbreeder: Self-Referential Self-Improvement Via Prompt Evolution" (<https://arxiv.org/abs/2309.16797>)
- Wei et al '22: Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models" (<https://arxiv.org/abs/2201.11903>)
- Kojima et al '23: Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, Yusuke Iwasawa, "Large Language Models are Zero-Shot Reasoners" (<https://arxiv.org/abs/2205.11916>)
- Fu et al '23: Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, Tushar Khot, "Complexity-Based Prompting for Multi-Step Reasoning" (<https://arxiv.org/abs/2210.00720>)
- Yao et al '23: Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, Karthik Narasimhan, "Tree of Thoughts: Deliberate Problem Solving with Large Language Models" (<https://arxiv.org/abs/2305.10601>)
- Wei et al '22: Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, William Fedus, "Emergent Abilities of Large Language Models", (<https://arxiv.org/abs/2206.07682>)
- Wang et al '22: Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, Huan Sun, "Towards Understanding Chain-of-Thought Prompting: An Empirical Study of What Matters", (<https://arxiv.org/abs/2212.10001>)
- Shi et al '22: Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, Jason Wei, "Language Models are Multilingual Chain-of-Thought Reasoners", (<https://arxiv.org/abs/2210.03057>)
- Zhang et al '23: Zhuosheng Zhang, Aston Zhang, Mu Li, Alex Smola, "Automatic Chain of Thought Prompting in Large Language Models" (<https://openreview.net/forum?id=5NTt8GFjUHkr>)
- Sprague et al '24: Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, Greg Durrett "To CoT or not to CoT? Chain-of-thought helps mainly on math and symbolic reasoning" (<https://arxiv.org/abs/2409.12183>)
- Gao et al '23: Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, Graham Neubig, "PAL: Program-aided Language Models" (<https://arxiv.org/abs/2211.10435>)
- Chen et al '22: Wenhui Chen, Xueguang Ma, Xinyi Wang, William W. Cohen, "Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks" (<https://arxiv.org/abs/2211.12588>)
- Schick et al '23: Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, Thomas Scialom, "Toolformer: Language Models Can Teach Themselves to Use Tools" (<https://arxiv.org/abs/2302.04761>)



Thank You!