CS 639: Foundation Models
Specialization: Fine-Tuning & Adaptation I

Fred Sala

University of Wisconsin-Madison

March 10, 2026

Announcements

- Midterm: **March 11, 5:40 pm - 7:20 pm**
  - Ingraham Hall, Room B10
- **Homework 2**: due today!
  - HW 3 out tonight
- **Project information**: released

https://pages.cs.wisc.edu/~fredsala/cs639/files/project_info_639.pdf

Tuesday March 10	Specialization I
Thursday March 12	Specialization II
Tuesday March 15	Alignment I
Thursday March 17	Alignment II

Modifying and Improving FMs

Announcements

- Midterm info: **March 11, 5:40 pm - 7:20 pm**
  - Main room: Ingraham Hall, Room B10
  - **McBurney: Morgridge Hall, Room B2556**
    - (Note: new location!)
  - Sample problems with solutions out.
  - **Format:** 20-25 questions, all multiple choice, some calculation, some conceptual
  - **Covers:** Up to Thursday's lecture (14)
  - **Cheat-sheet:** allowed. 1 page, both sides, handwritten

Outline

- **From Last Time: Improving + Extending Prompting**
  - Searching for good prompts, techniques for continuous/soft prompts, ensembling, prompt optimization
- **Chain-of-Thought**
  - CoT basics, generalizations, variations
- **Fine-Tuning and Adapters Intro**
  - Fine-tuning vs. prompting, linear probing, etc. Full vs partial fine tuning vs adapting. Popular adapters

Outline

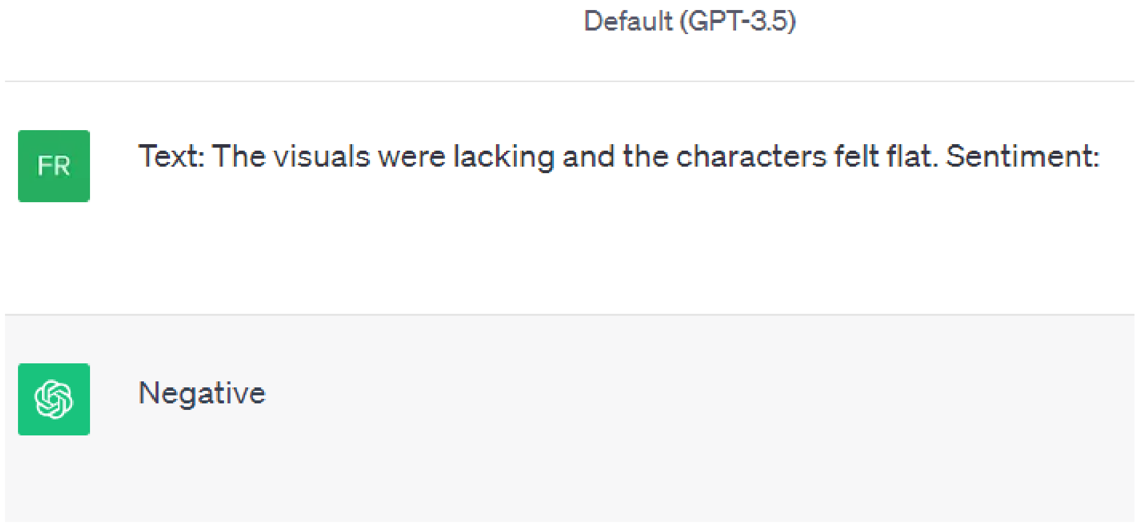
- **From Last Time: Improving + Extending Prompting**
  - Searching for good prompts, techniques for continuous/soft prompts, ensembling, prompt optimization
- **Chain-of-Thought**
  - CoT basics, generalizations, variations
- **Fine-Tuning and Adapters Intro**
  - Fine-tuning vs. prompting, linear probing, etc. Full vs partial fine tuning vs adapting. Popular adapters

Recall: Prompting---Ask Your Model

Essentially, ask your model to perform your goal task

Example: sentiment analysis task

- Prompt: "`Text: The visuals were lacking and the characters felt flat. Sentiment:`"
- Result: "`Negative`"



Text: The visuals were lacking and the characters felt flat. Sentiment:

Negative

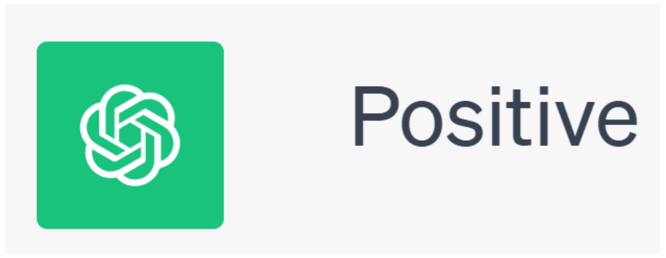
Recall: Prompting---Zero-shot vs Few-shot

Terminology:

- **Zero-shot:** No “examples” provided to the model.
- **Few-shot/in-context learning (ICL)**: Provide “examples”

```
Input: Subpar acting.   Sentiment: Negative
Input: Beautiful film.  Sentiment: Positive
Input: Amazing.         Sentiment:
```

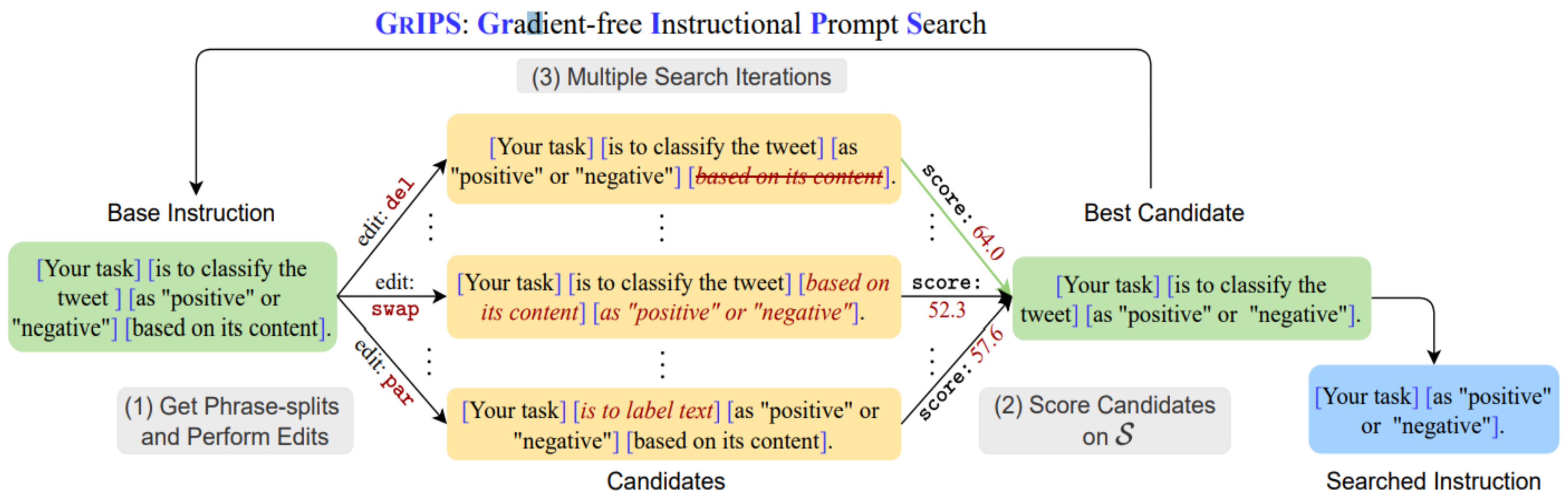
Zhao et al '21


Positive

Hard Prompting: **Discrete Optimization**

Goal: obtain a good prompt (e.g., an instruction)

- Avoiding gradients, a variety of search approaches

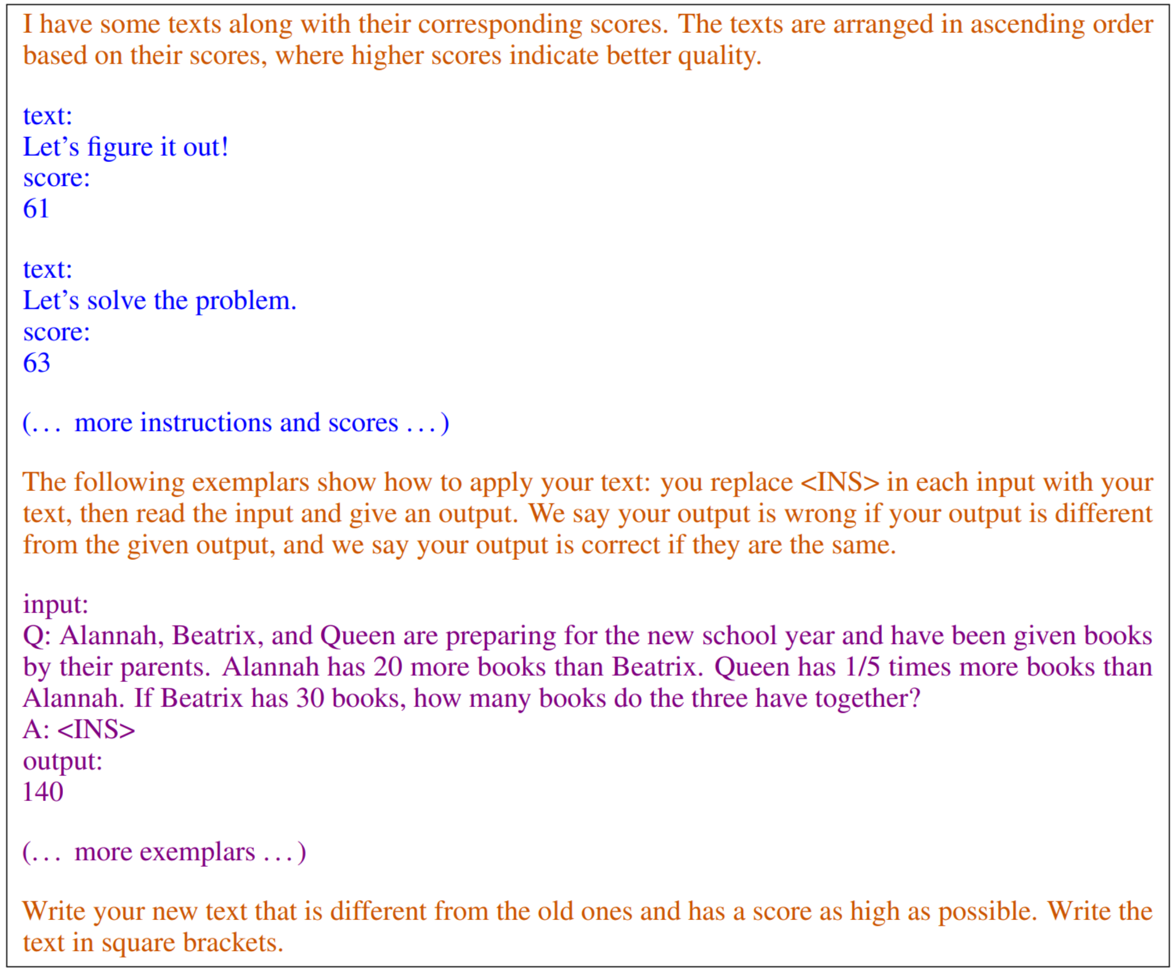

GRIPS: Gradient-free Instructional Prompt Search
(3) Multiple Search Iterations
Base Instruction
[Your task] [is to classify the tweet] [as "positive" or "negative"] [based on its content].
edit: del
edit: swap
edit: par
[Your task] [is to classify the tweet] [as "positive" or "negative"] [~~based on its content~~].
[Your task] [is to classify the tweet] [based on its content] [as "positive" or "negative"].
[Your task] [is to label text] [as "positive" or "negative"] [based on its content].
Candidates
score: 64.0
score: 52.3
score: 57.6
Best Candidate
[Your task] [is to classify the tweet] [as "positive" or "negative"].
(1) Get Phrase-splits and Perform Edits
(2) Score Candidates on $\mathcal{S}$
[Your task] [as "positive" or "negative"].
Searched Instruction

Prasad et al '23

LLMs as Optimizers: Example

Recall our goal: obtain good instructions for an LLM

- Want: use ***another*** LLM as an optimizer



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 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.

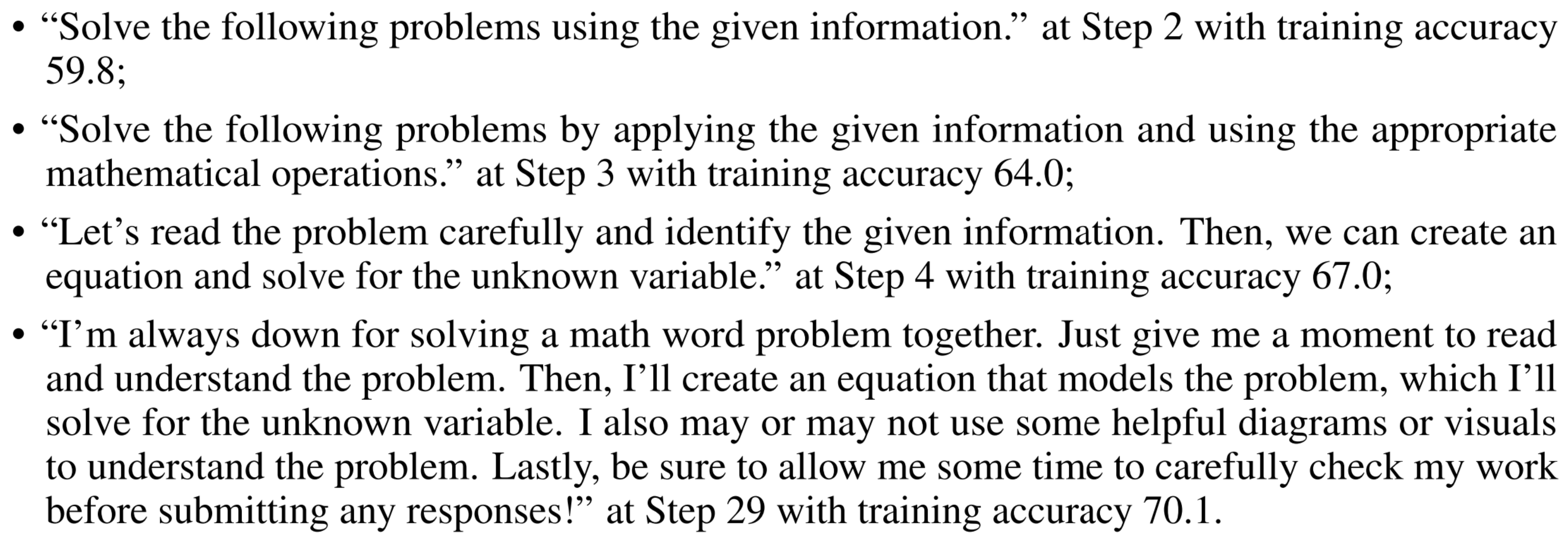
Meta-instructions

Trajectory points

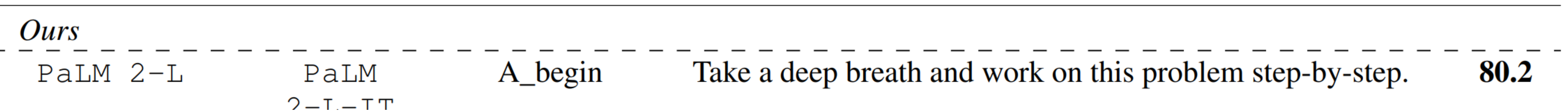
Problem to be solved

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-L-IT`	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

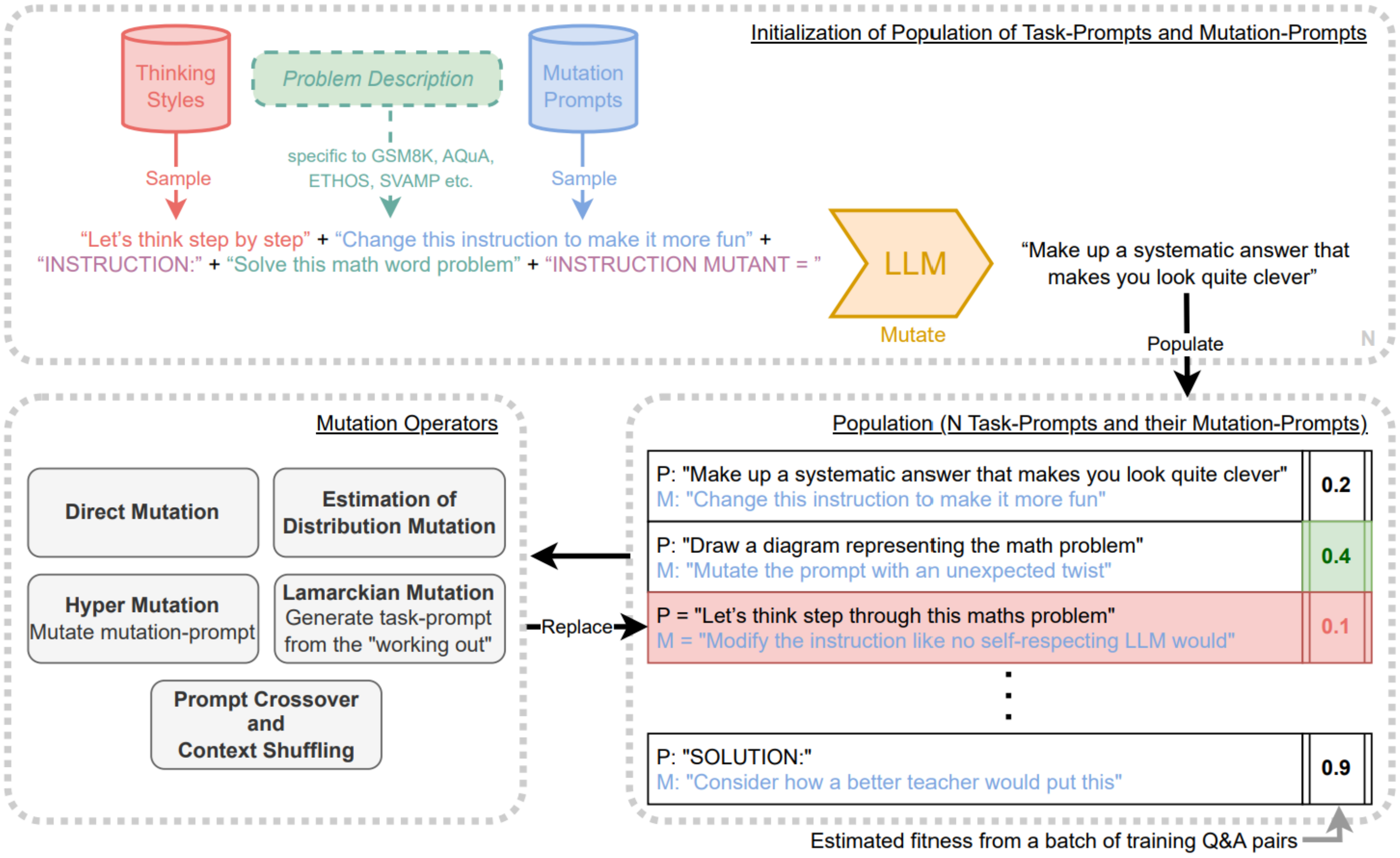
- Lots of classic search methods only require notion of “**neighbors**” and **evaluation** function access
    - Hill-climbing
    - Simulated annealing
    - Genetic algorithms

- “**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

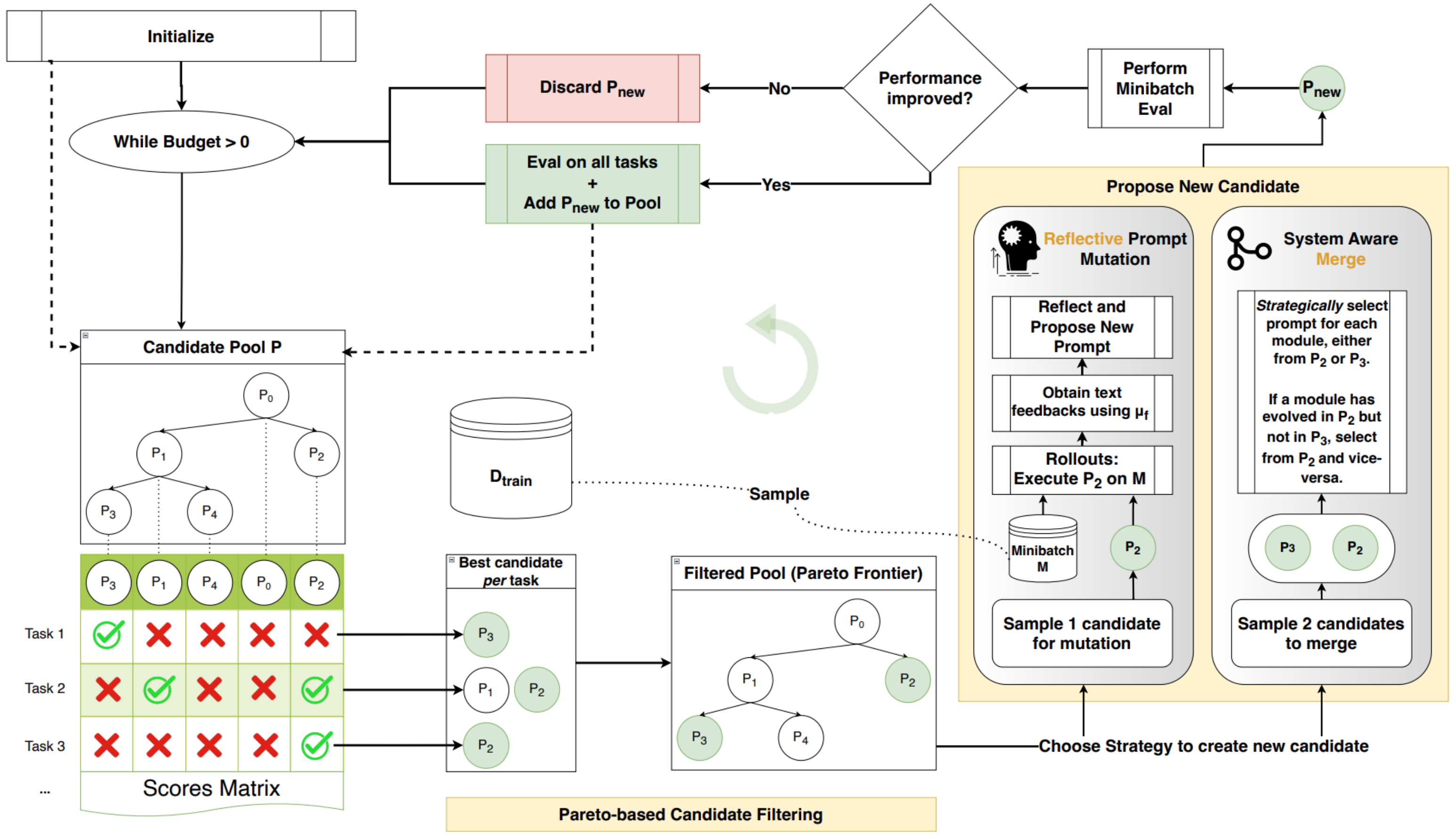


Initialization of Population of Task-Prompts and Mutation-Prompts
Thinking Styles
Problem Description
Mutation Prompts
Sample
specific to GSM8K, AQuA, ETHOS, SVAMP etc.
Sample
"Let's think step by step" + "Change this instruction to make it more fun" +
"INSTRUCTION:" + "Solve this math word problem" + "INSTRUCTION MUTANT = "
LLM
Mutate
"Make up a systematic answer that makes you look quite clever"
Populate
N
Mutation Operators
Direct Mutation
Estimation of Distribution Mutation
Hyper Mutation
Mutate mutation-prompt
Lamarckian Mutation
Generate task-prompt from the "working out"
Prompt Crossover and Context Shuffling
Replace
Population (N Task-Prompts and their Mutation-Prompts)
P: "Make up a systematic answer that makes you look quite clever"
M: "Change this instruction to make it more fun"
0.2
P: "Draw a diagram representing the math problem"
M: "Mutate the prompt with an unexpected twist"
0.4
P = "Let's think step through this maths problem"
M = "Modify the instruction like no self-respecting LLM would"
0.1
P: "SOLUTION:"
M: "Consider how a better teacher would put this"
0.9
Estimated fitness from a batch of training Q&A pairs

Fernando et al '23

More Auto-Prompting Work

“**GEPA**”: genetic algorithms + reflective iteration

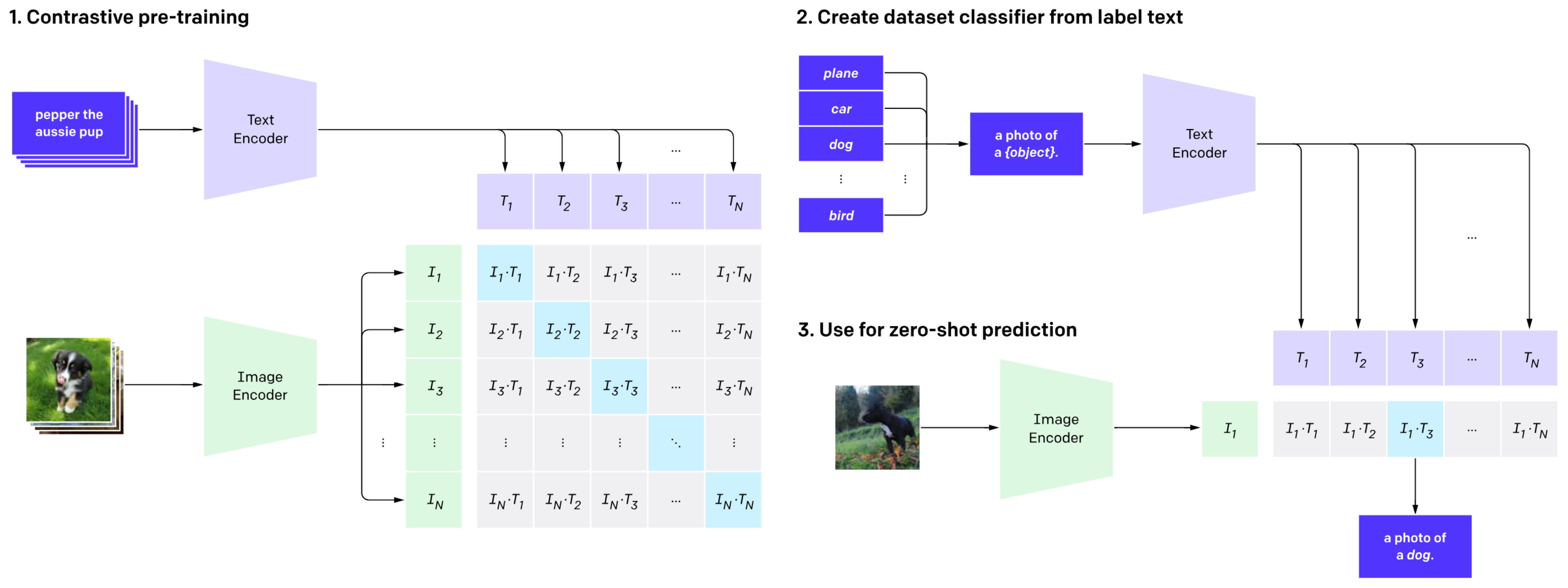


Initialize
While Budget > 0
Discard P_new
Eval on all tasks + Add P_new to Pool
No
Yes
Performance improved?
Perform Minibatch Eval
P_new
Propose New Candidate
Reflective Prompt Mutation
Reflect and Propose New Prompt
Obtain text feedbacks using μf
Rollouts: Execute P2 on M
Minibatch M
Sample 1 candidate for mutation
System Aware Merge
Strategically select prompt for each module, either from P2 or P3. If a module has evolved in P2 but not in P3, select from P2 and vice-versa.
Sample 2 candidates to merge
Candidate Pool P
D_train
Sample
Best candidate per task
Filtered Pool (Pareto Frontier)
Task 1
Task 2
Task 3
...
Scores Matrix
Choose Strategy to create new candidate
Pareto-based Candidate Filtering

Agrawal et al ‘26

Prompting VLMs and Multimodal Models

Recall training and prediction in CLIP-style VLMs



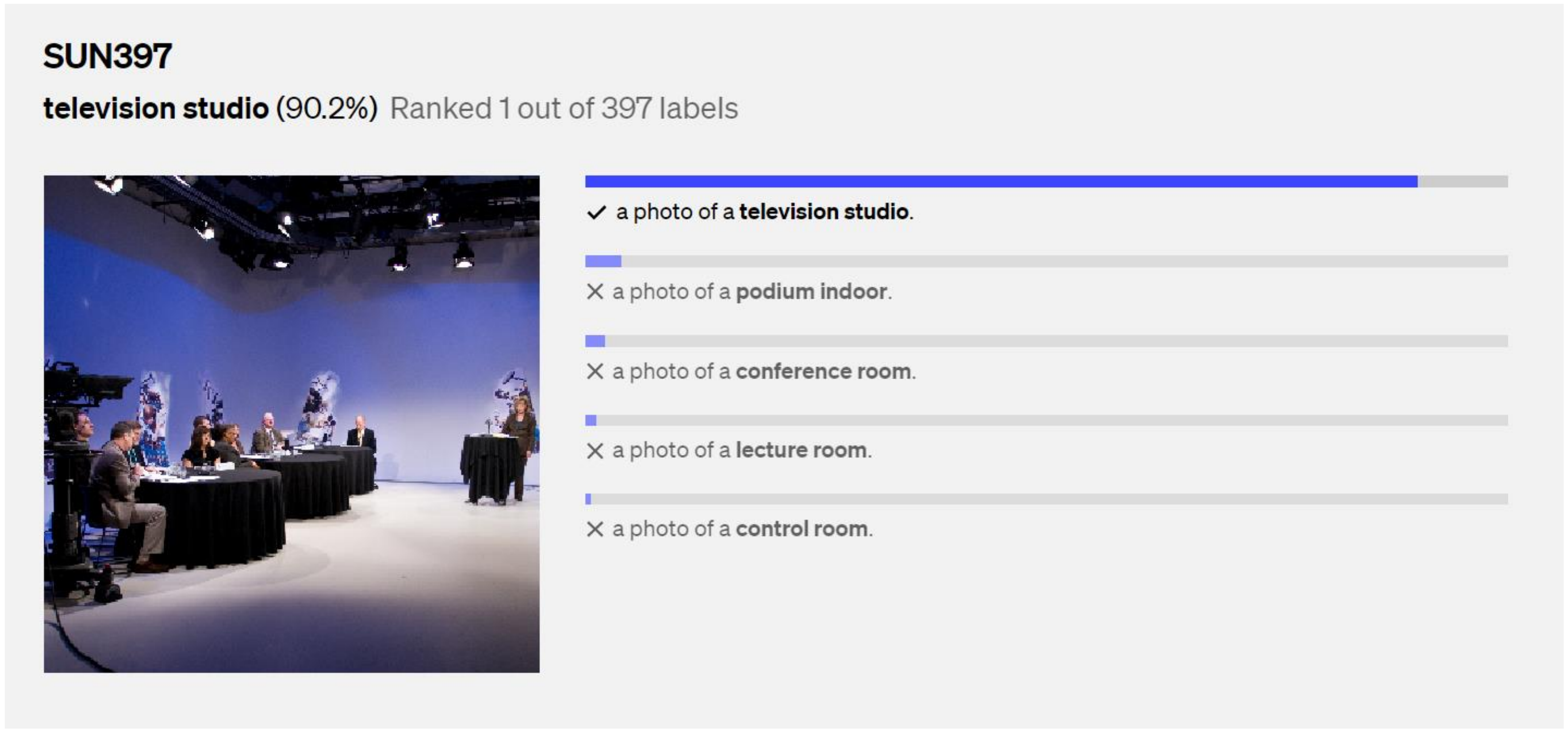
1. Contrastive pre-training
pepper the aussie pup
Text Encoder
Image Encoder
T_1 T_2 T_3 ... T_N
I_1 I_2 I_3 ⋮ I_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
$I_2 \cdot T_1$ $I_2 \cdot T_2$ $I_2 \cdot T_3$ ... $I_2 \cdot T_N$
$I_3 \cdot T_1$ $I_3 \cdot T_2$ $I_3 \cdot T_3$ ... $I_3 \cdot T_N$
$I_N \cdot T_1$ $I_N \cdot T_2$ $I_N \cdot T_3$ ... $I_N \cdot T_N$
2. Create dataset classifier from label text
plane
car
dog
bird
a photo of a {object}.
Text Encoder
3. Use for zero-shot prediction
Image Encoder
I_1
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
a photo of a dog.

OpenAI

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**.

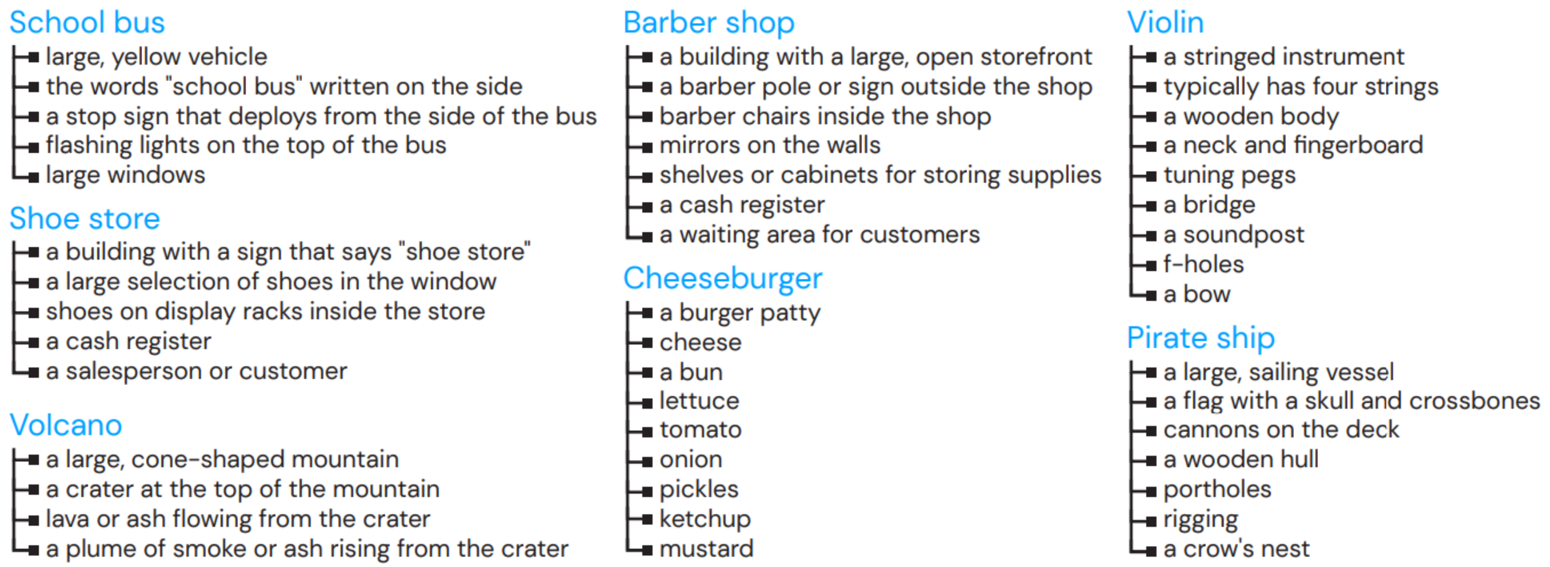
✗ a photo of a **control room**.

OpenAI

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
- 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

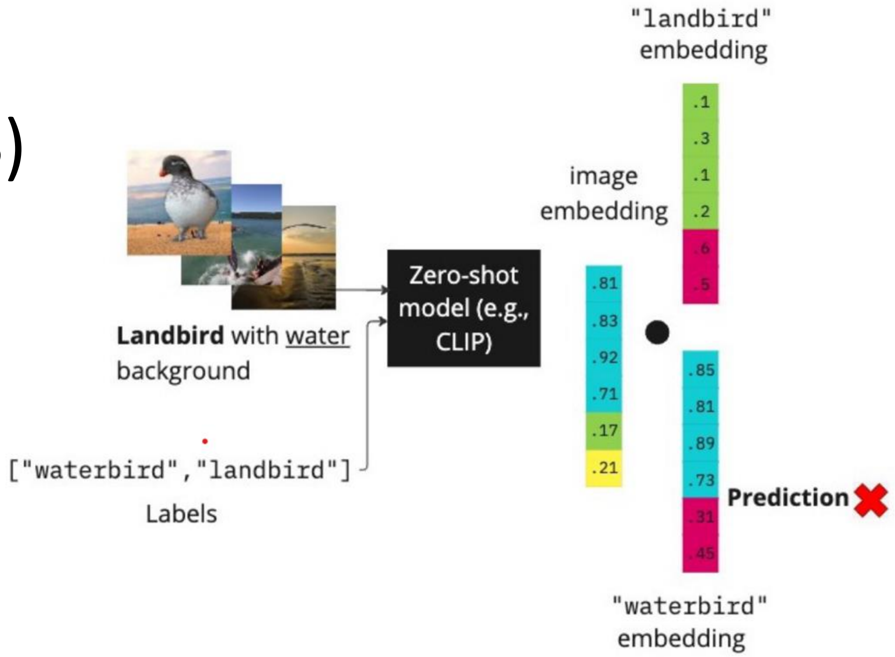
Figure 3: Examples of descriptor schema produced by GPT-3.

LLMs to Improve VLMs: **Spurious Features**

This helped with positives.

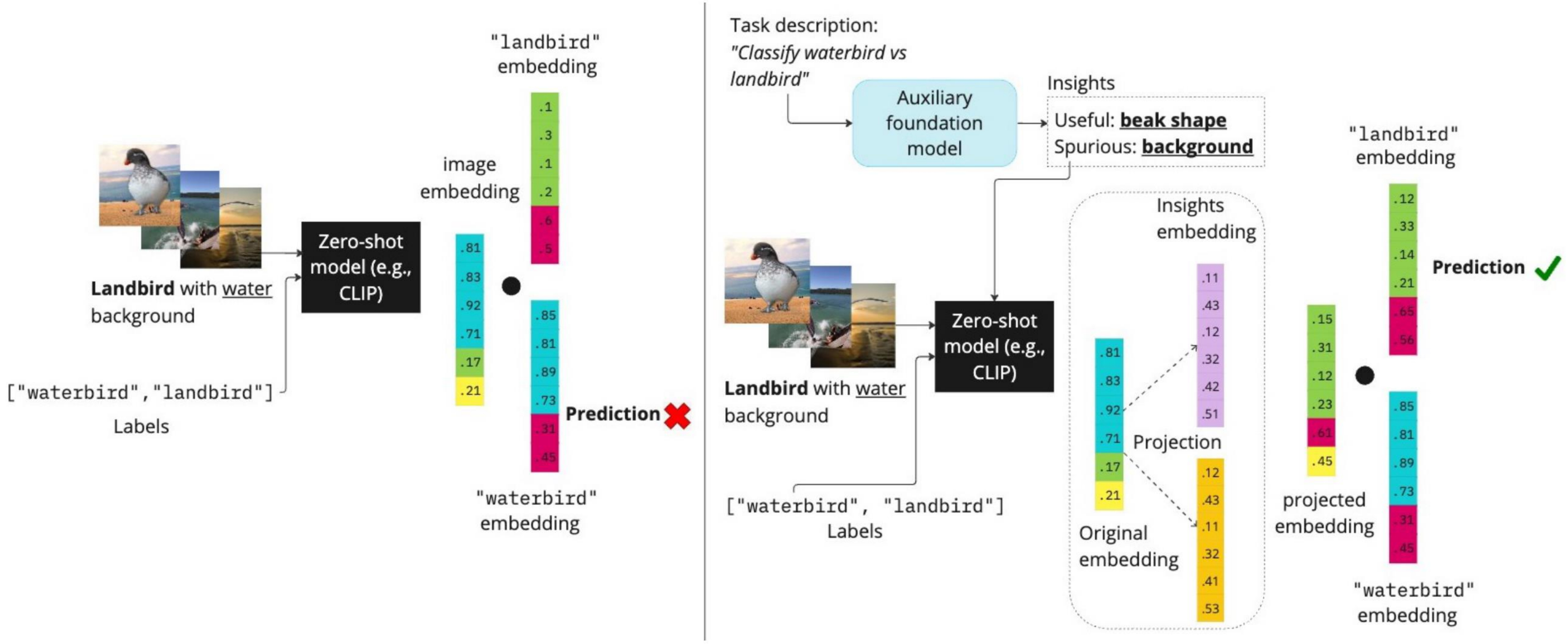
- What about **negatives** (i.e., spurious features?)
  - Example: waterbirds with CLIP

- Spurious correlations: generally a problem with all pretrained models
  - But LLMs can also tell us about this (Adila '23)



LLMs to Improve VLMs: **Spurious Features**

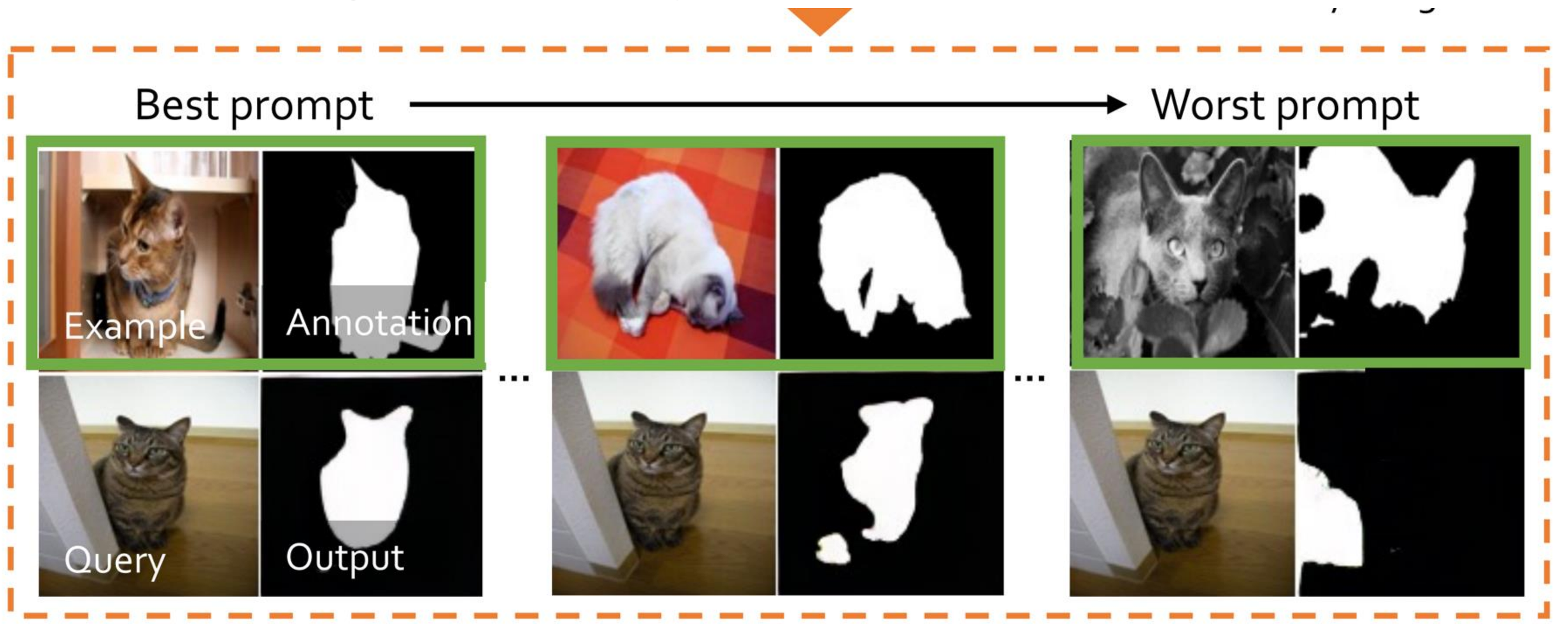
Modify embeddings used for prediction:



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

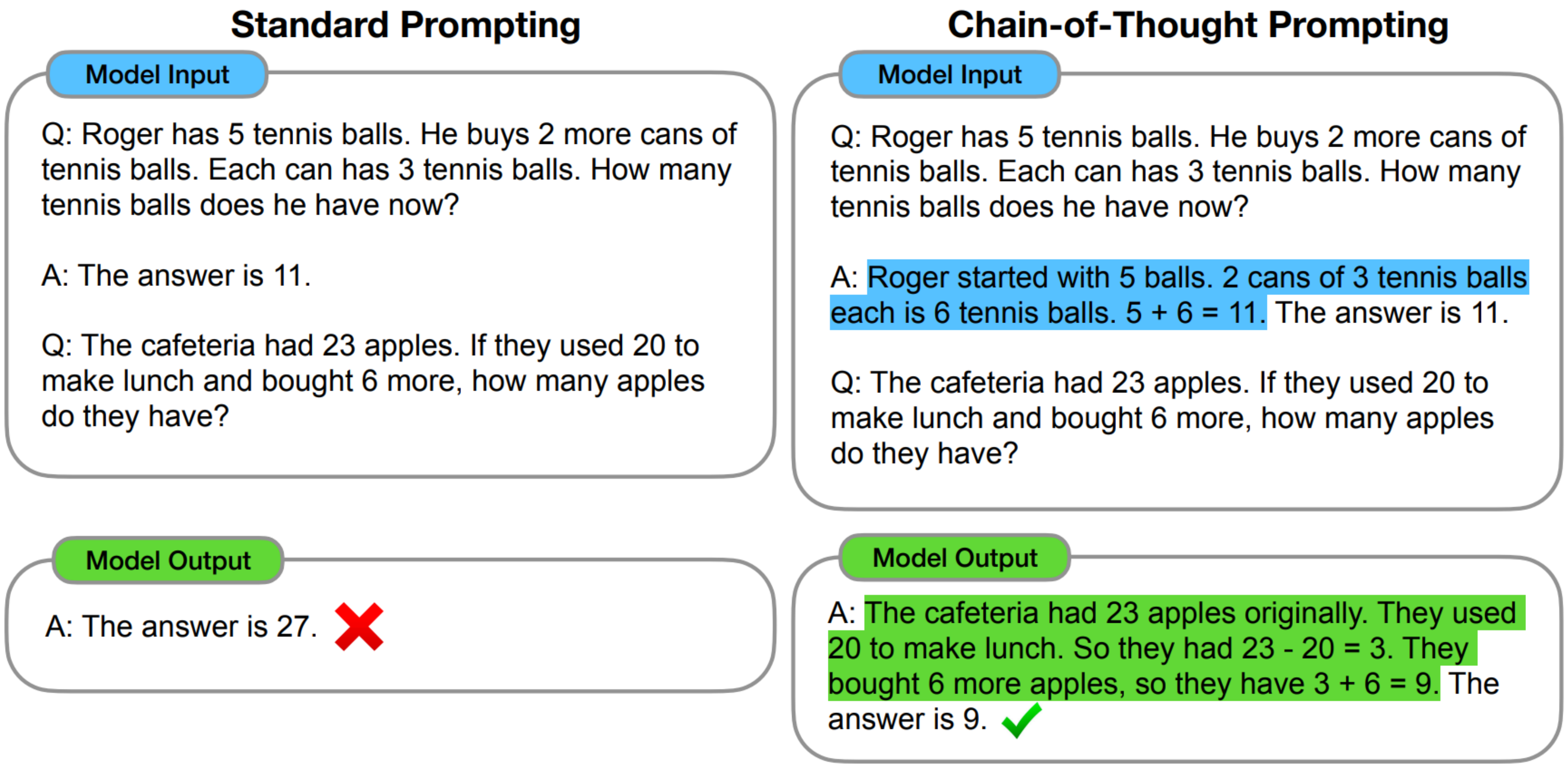
Break & Questions

Outline

- **From Last Time: Improving + Extending Prompting**
  - Searching for good prompts, techniques for continuous/soft prompts, ensembling, prompt optimization
- **Chain-of-Thought**
  - CoT basics, generalizations, variations
- **Fine-Tuning and Adapters Intro**
  - Fine-tuning vs. prompting, linear probing, etc. Full vs partial fine tuning vs adapting. Popular adapters

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

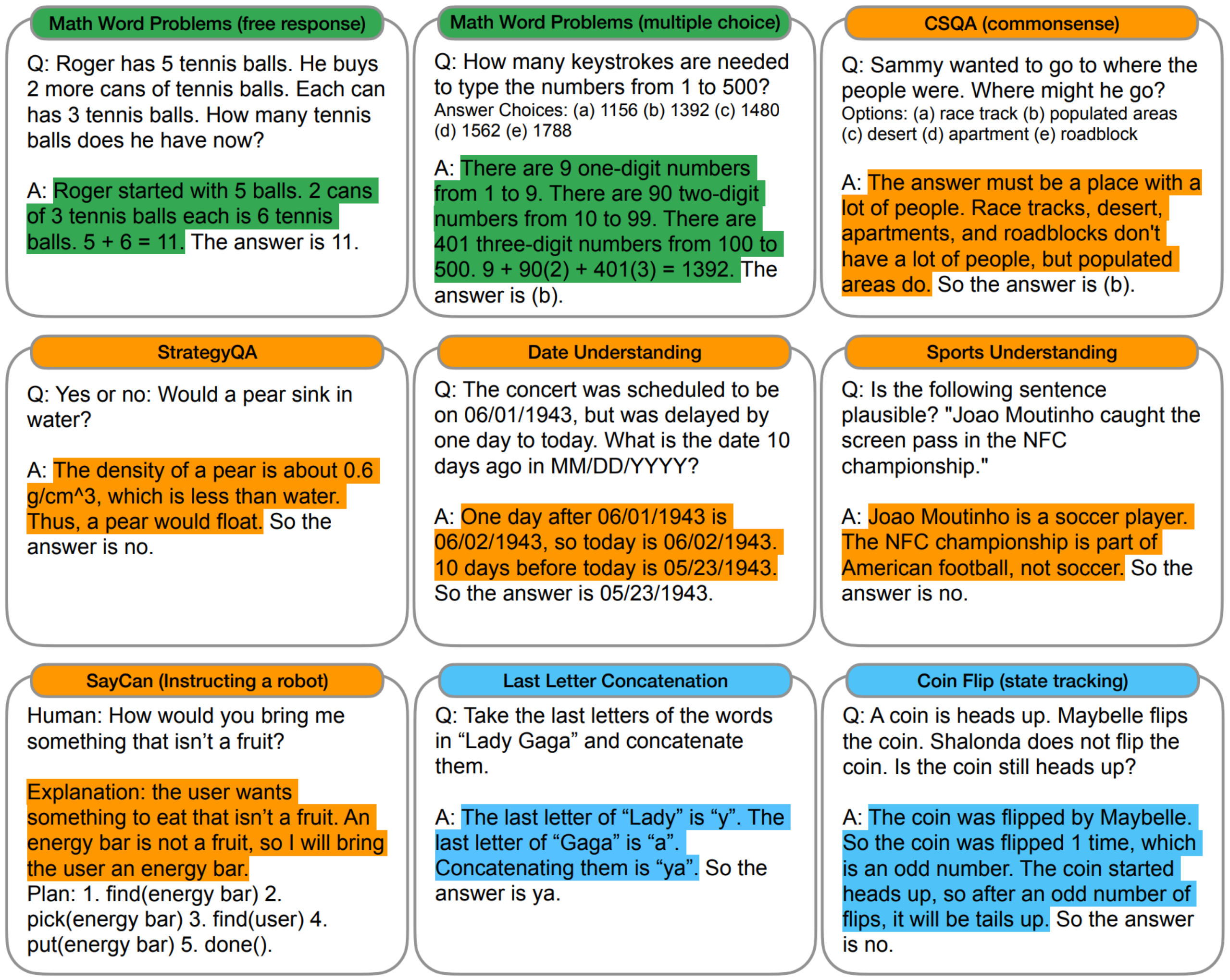
A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Wei et al '22

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)

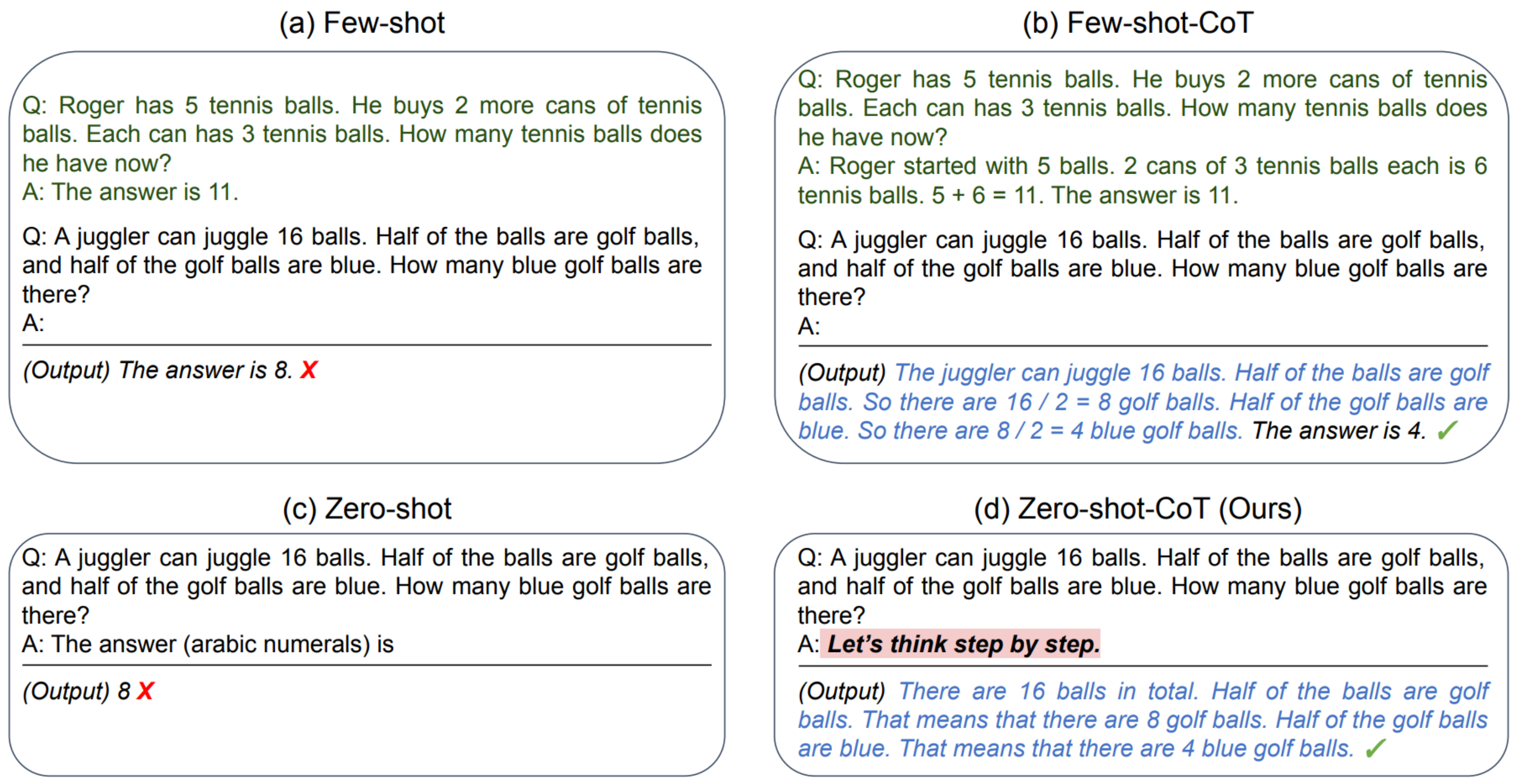
Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up?

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. ***X***

(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 ***X***

(d) Zero-shot-CoT (Ours)

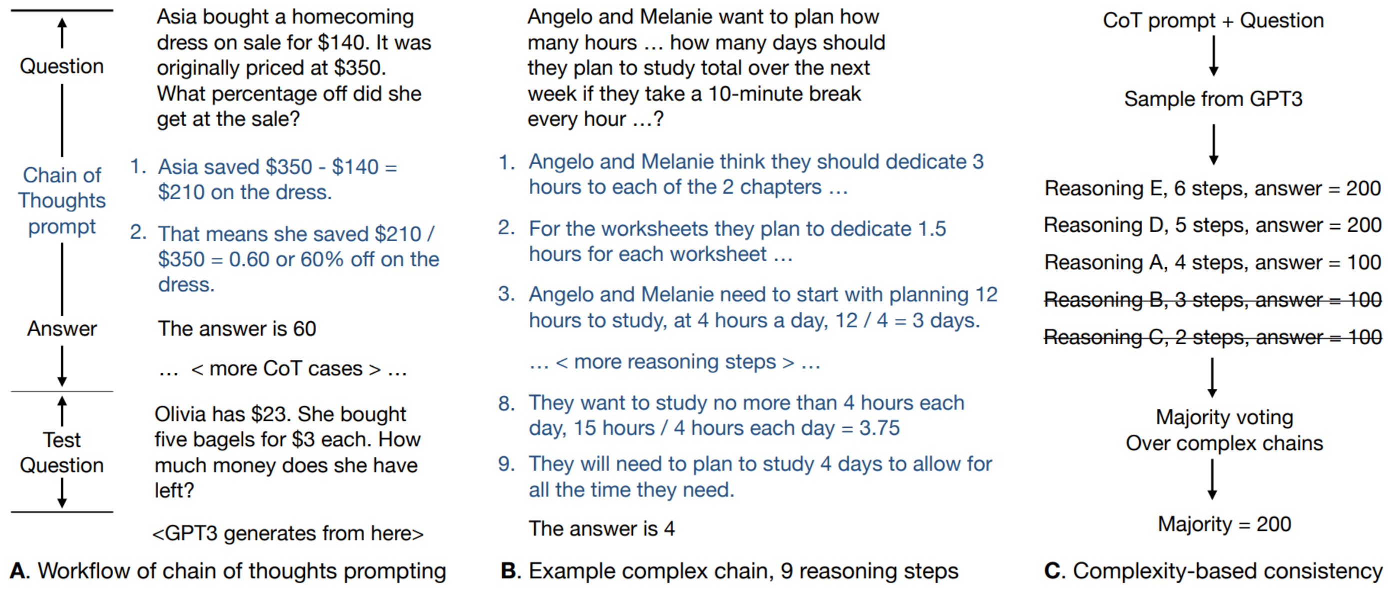
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let’s think step by step.***

(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.



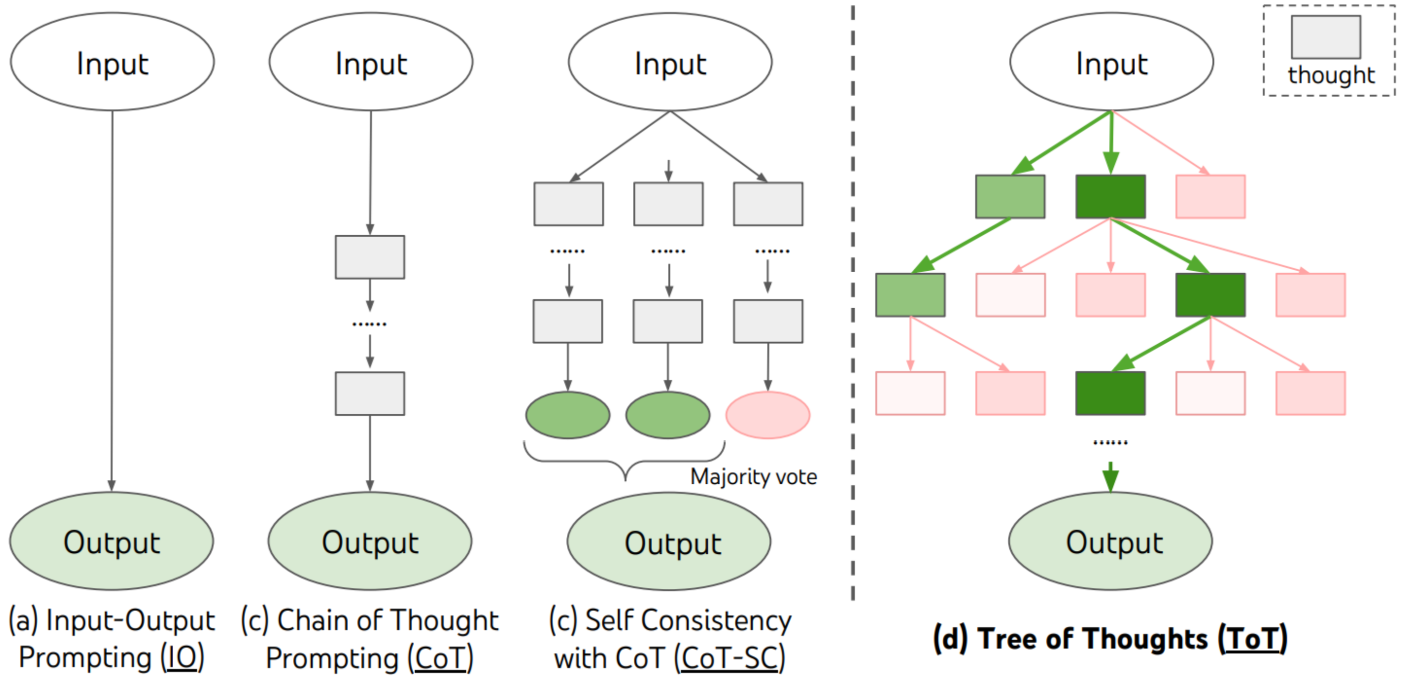
Question
Chain of Thoughts prompt
Answer
Test Question
Asia bought a homecoming dress on sale for $140. It was originally priced at $350. What percentage off did she get at the sale?
1. Asia saved $350 - $140 = $210 on the dress.
2. That means she saved $210 / $350 = 0.60 or 60% off on the dress.
The answer is 60
… < more CoT cases > …
Olivia has $23. She bought five bagels for $3 each. How much money does she have left?
<GPT3 generates from here>
A. Workflow of chain of thoughts prompting
Angelo and Melanie want to plan how many hours … how many days should they plan to study total over the next week if they take a 10-minute break every hour …?
1. Angelo and Melanie think they should dedicate 3 hours to each of the 2 chapters …
2. For the worksheets they plan to dedicate 1.5 hours for each worksheet …
3. Angelo and Melanie need to start with planning 12 hours to study, at 4 hours a day, 12 / 4 = 3 days.
… < more reasoning steps > …
8. They want to study no more than 4 hours each day, 15 hours / 4 hours each day = 3.75
9. They will need to plan to study 4 days to allow for all the time they need.
The answer is 4
B. Example complex chain, 9 reasoning steps
CoT prompt + Question
Sample from GPT3
Reasoning E, 6 steps, answer = 200
Reasoning D, 5 steps, answer = 200
Reasoning A, 4 steps, answer = 100
~~Reasoning B, 3 steps, answer = 100~~
~~Reasoning C, 2 steps, answer = 100~~
Majority voting Over complex chains
Majority = 200
C. Complexity-based consistency

Fu et al '23

Chain-of-Thought: **Generalizations**

How do we really "reason"?

- Not really by sampling a bunch of chains…


Input
Input
Input
Input
thought
Majority vote
Output
Output
Output
Output
(a) Input-Output Prompting (IO)
(c) Chain of Thought Prompting (CoT)
(c) Self Consistency with CoT (CoT-SC)
(d) Tree of Thoughts (ToT)

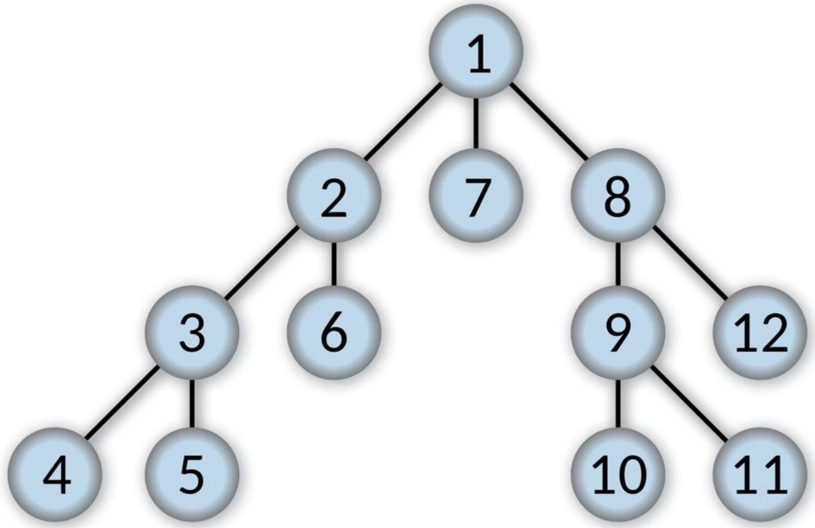
Yao et al '23

Chain-of-Thought: **Generalizations**

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

- **Notation**: thoughts $z_1, z_2, \ldots, z_n$ bridge x and y
- Comparison to other methods:
  - Vanilla CoT: sample $z_i \sim p_\theta(z_i \mid x, z_1,\ldots,z_{i-1})$, $y \sim p_\theta(y \mid x, z_1,\ldots,z_n)$
  - CoT Self-Consistency: sample multiple times, take majority vote

- Idea: create a state $s=[x,z_1,\ldots,z_n]$
- Generate multiple candidates for next state
  - Then run standard search (i.e., BFS, DFS, A*)

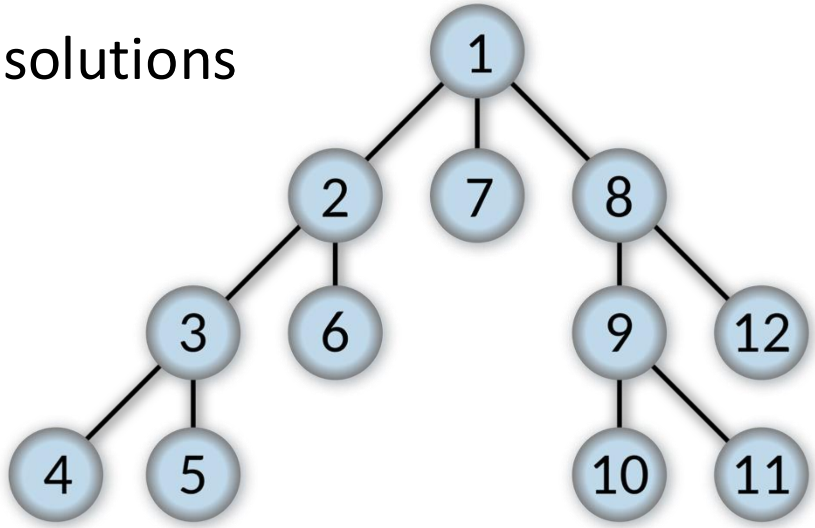


Drichel (Wiki)

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



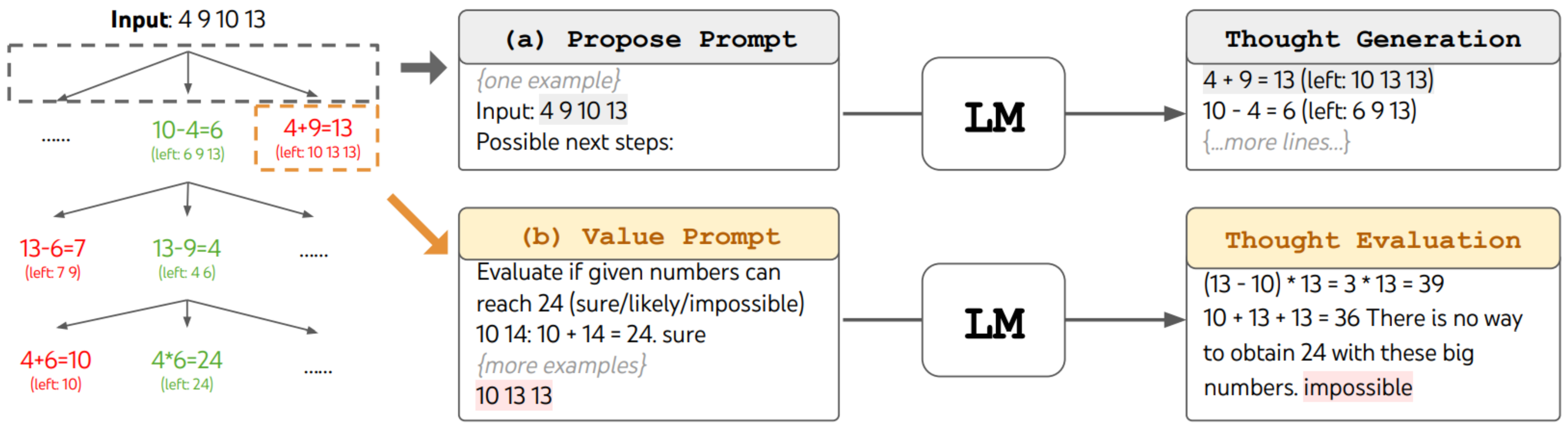
Drichel (Wiki)

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 basi
arithmetic operations (+-*/) to obtain 24. For example, given input "4 9 10 13", a solution outpu
could be "(10 - 4) * (13 - 9) = 24".



(a) Propose Prompt
{one example}
Input: 4 9 10 13
Possible next steps:

Thought Generation
4 + 9 = 13 (left: 10 13 13)
10 - 4 = 6 (left: 6 9 13)
{...more lines...}

(b) Value Prompt
Evaluate if given numbers can reach 24 (sure/likely/impossible)
10 14: 10 + 14 = 24. sure
{more examples}
10 13 13

Thought Evaluation
(13 - 10) * 13 = 3 * 13 = 39
10 + 13 + 13 = 36 There is no way to obtain 24 with these big numbers. impossible

Break & Questions

Outline

- **From Last Time: Improving + Extending Prompting**
  - Searching for good prompts, techniques for continuous/soft prompts, ensembling, prompt optimization
- **Chain-of-Thought**
  - CoT basics, generalizations, variations
- **Fine-Tuning and Adapters Intro**
  - Fine-tuning vs. prompting, linear probing, etc. Full vs partial fine tuning vs adapting. Popular adapters

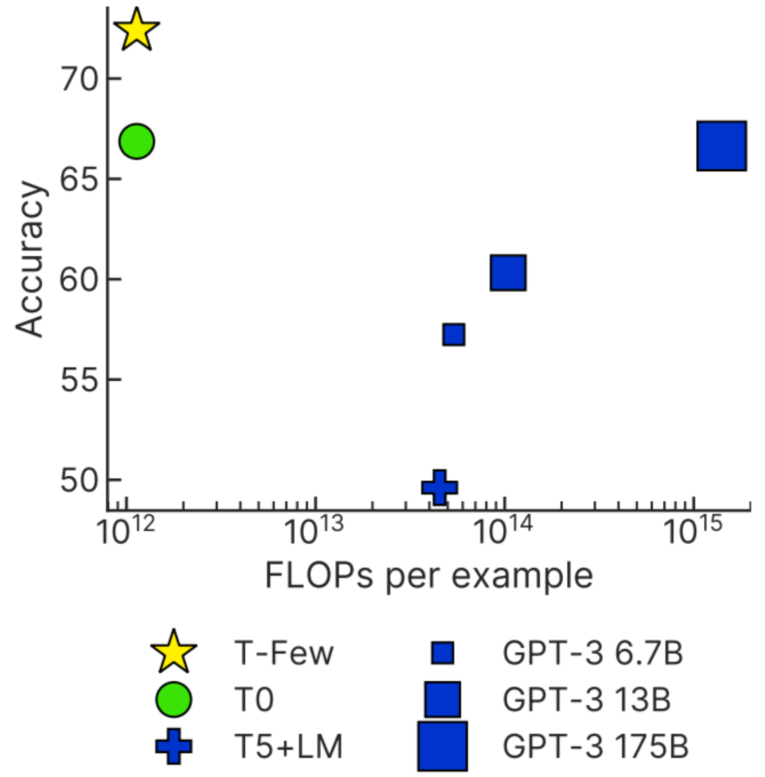
Before: **Prompting**

With prompting, we didn't change the model

- To improve performance, we used few-shot/ICL
- But, this might be **worse** than changing our model weights

Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning

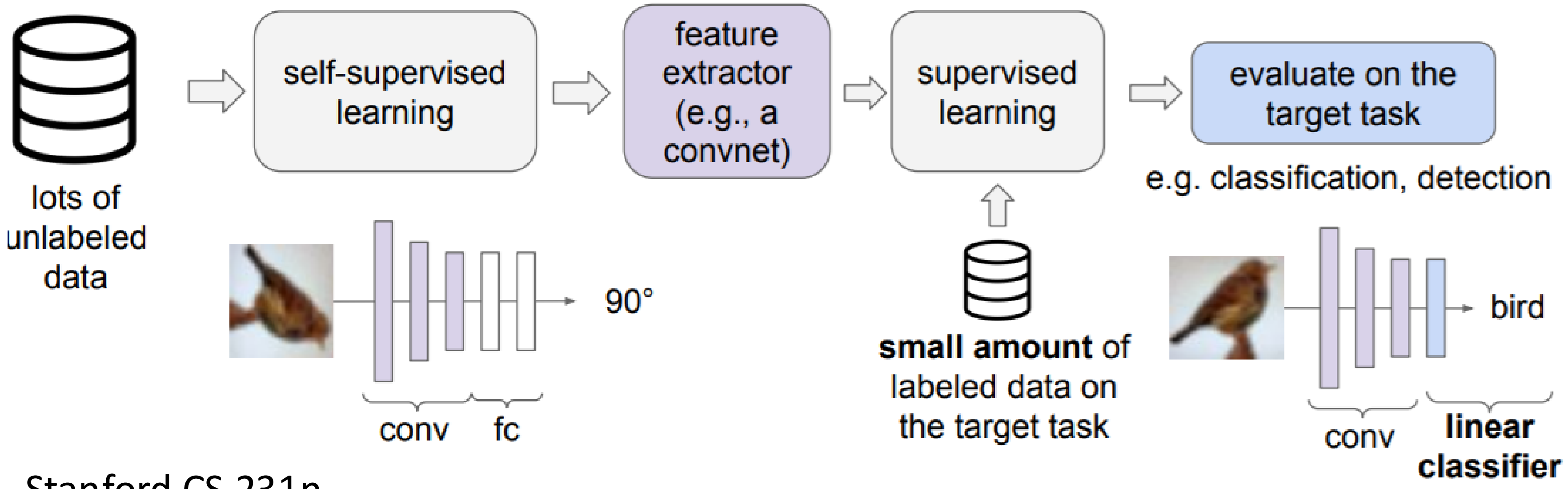
Liu et al '22



Before: **Frozen Models/Linear Probing**

We previously discussed freezing our model, and using just some trainable heads

- E.g., a linear model on top (called **linear probing**)
- Our self-supervised learning example



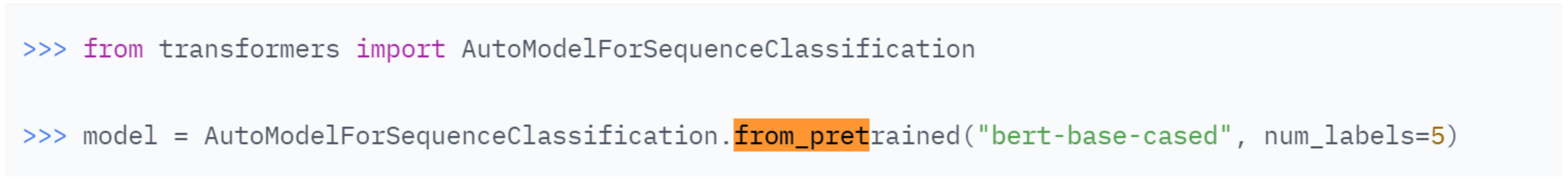
lots of unlabeled data
self-supervised learning
feature extractor (e.g., a convnet)
supervised learning
evaluate on the target task
e.g. classification, detection
90°
conv
fc
small amount of labeled data on the target task
bird
conv
linear classifier

Stanford CS 231n

Full Fine-Tuning

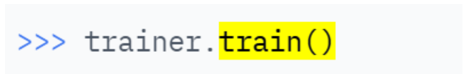
Performance might still be bottlenecked,

- Frozen representations might not be suitable for task
- Might need lots of capacity on top to adapt
- **Change all the weights!**



```
>>> from transformers import AutoModelForSequenceClassification

>>> model = AutoModelForSequenceClassification.from_pretrained("bert-base-cased", num_labels=5)
```



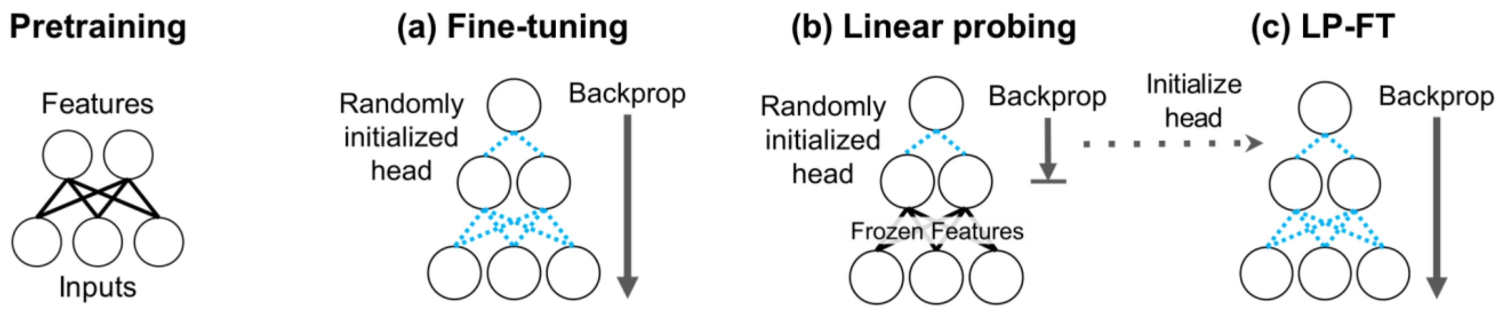
```
>>> trainer.train()
```

https://huggingface.co/docs/transformers/training

Full Fine-Tuning: **Downsides**

Fine-tuning all parameters can be tough:

1. **Expensive:** just like training a full model
2. **Known to cause issues on OOD data…**
   - Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution



	(a) Fine-tuning	(b) Linear probing	(c) LP-FT
ID test	85.1%	82.9%	**85.7%**
OOD test	59.3%	66.2%	**68.9%**

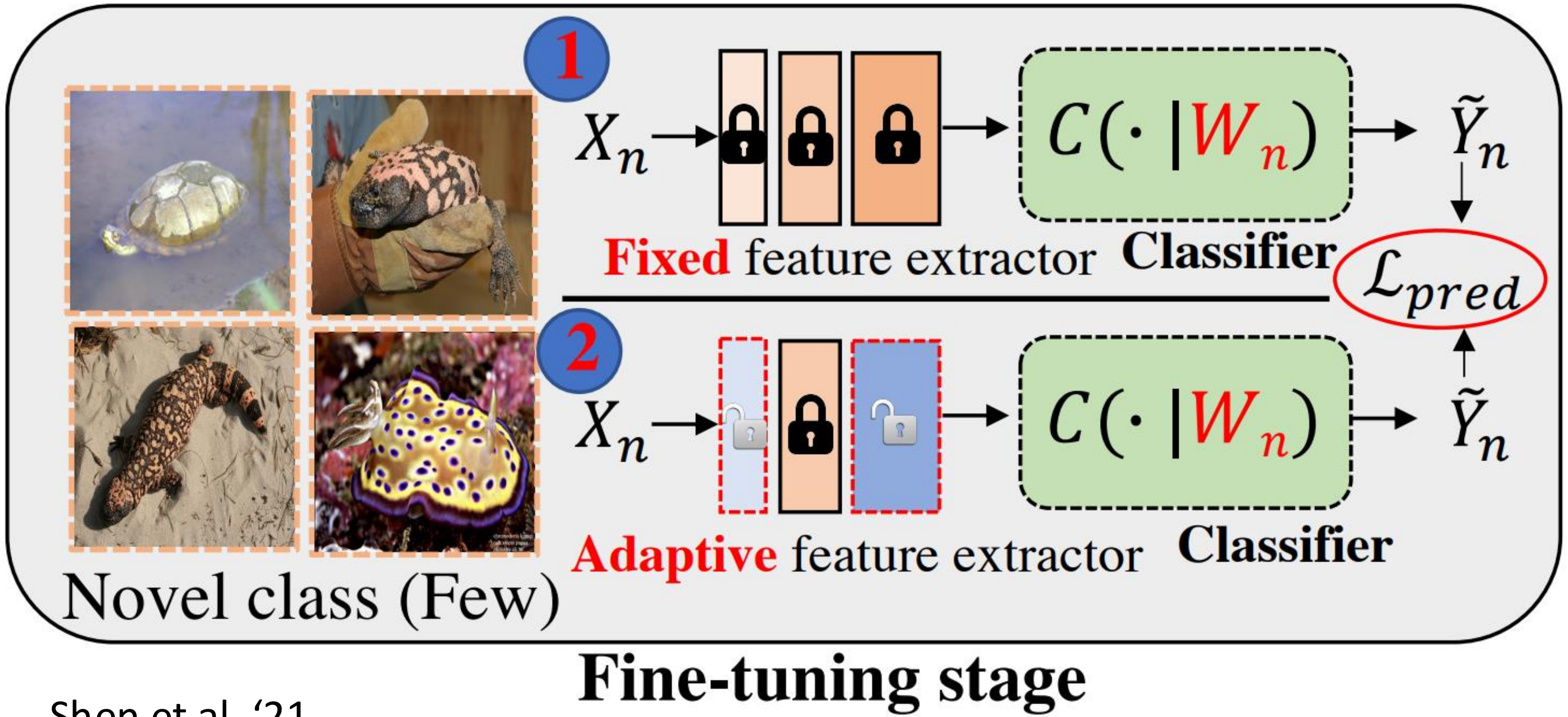
Average accuracies (10 distribution shifts)

Kumar et al '22

Partial Fine-Tuning

Full fine-tuning might be expensive

- Partial fine-tuning might be a good choice
- Only some layers change

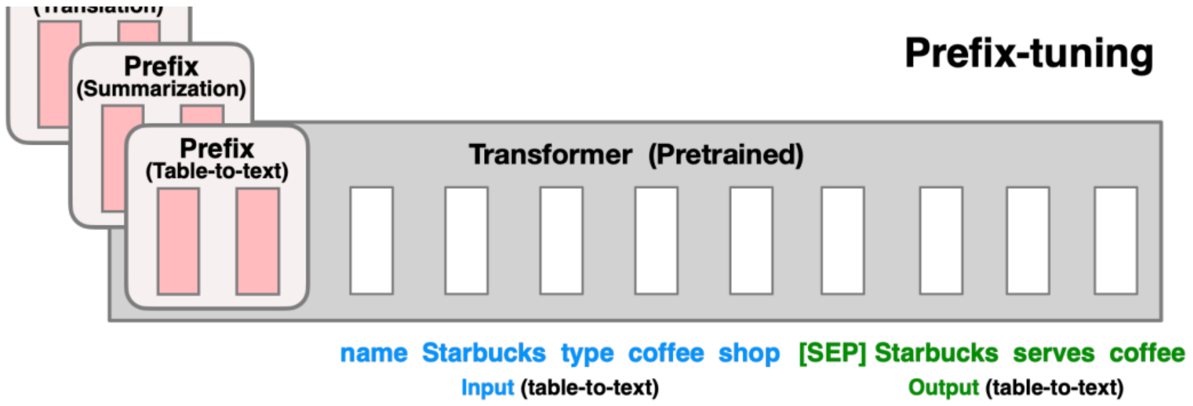


Shen et al, '21

Prefix-Tuning

Recall this *soft prompting* method.

- Prefixes are trainable parameters
- Train one for each goal task, only store these new parameters
- Enables cheap **adaptation** of frozen language model

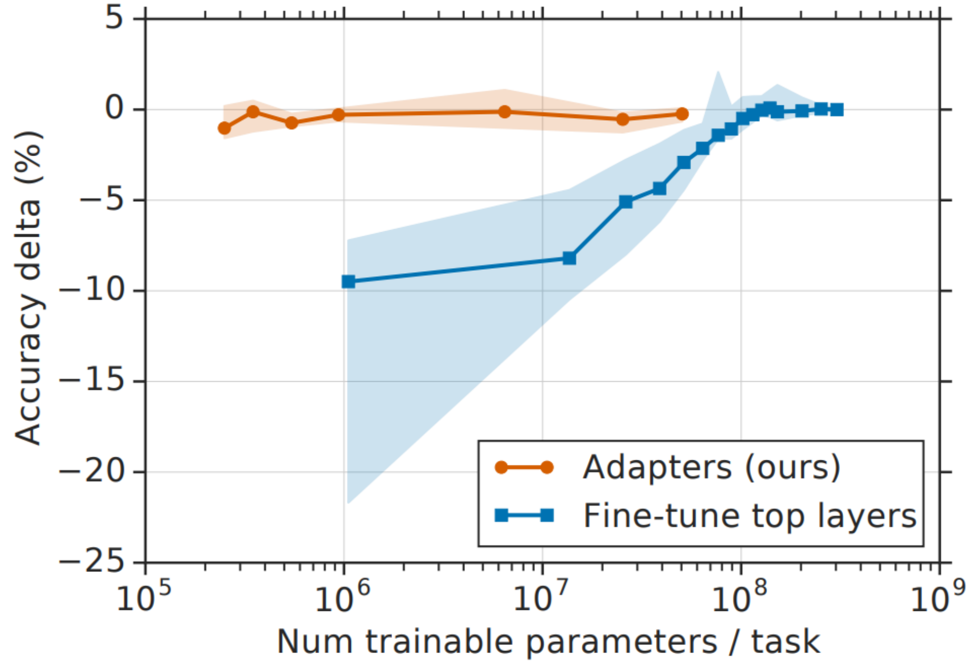


Li and Liang '21

Parameter-Efficient Fine-Tuning (PEFT)

None of these methods were full satisfying

- Have to figure out what layers to train, have to interpolate with prompts, etc.
  - Lots of choices!
- If we fine-tune too many parameters, that gets expensive…
  - But top only, performance isn't **great**
- Houlsby et al '19:



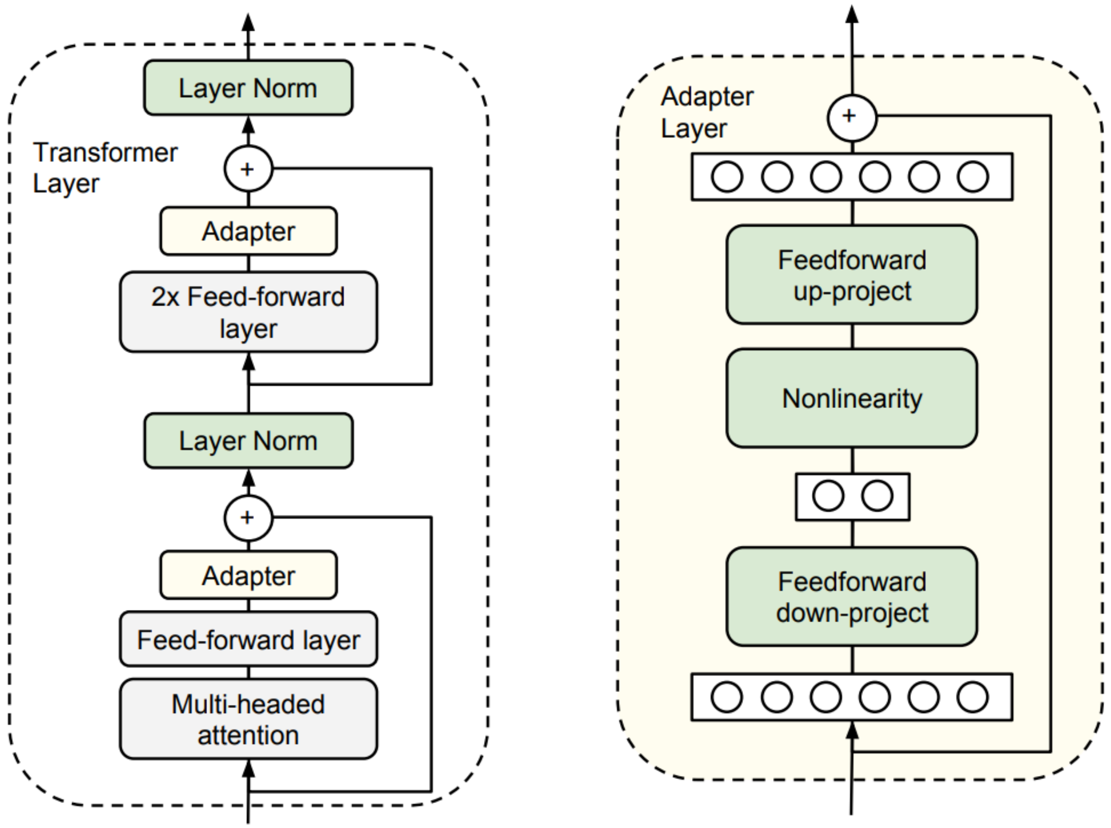
PEFT: **Adapters**

Want two things in PEFT

- Good performance (accuracy, etc.)
- Parameter efficiency

- Solution: **Adapters**
  - Small modules, inserted
  in between model and trained

Another **advantage:** no change to model, new modules for tasks



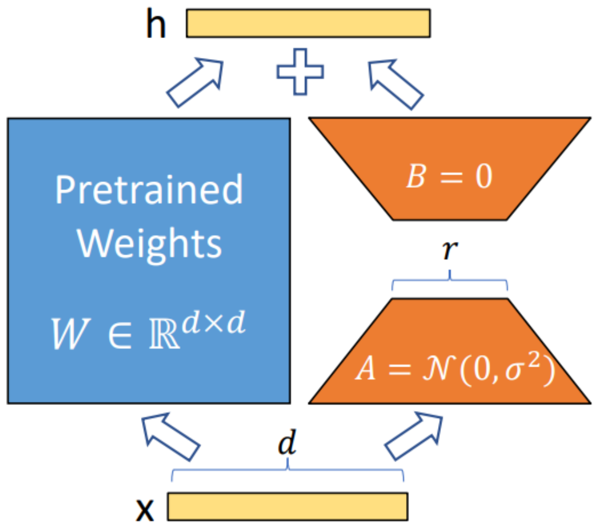
Houlsby et al '19

PEFT: **Low-Rank Adapters (LoRA)**

Perhaps the most popular variant

- LoRA suggests **adding** directly to pretrained weights
  - Instead of placing in a new module
  - The matrix to be added should be low-rank
  - Intuition: the weight matrices already live close to a low-rank manifold

- Transformers, initially applied only to a Attention weight matrices
- Now everywhere


h
Pretrained Weights
$W \in \mathbb{R}^{d \times d}$
$B = 0$
r
$A = \mathcal{N}(0, \sigma^2)$
d
x

Hu et al ‘22

Thank You!