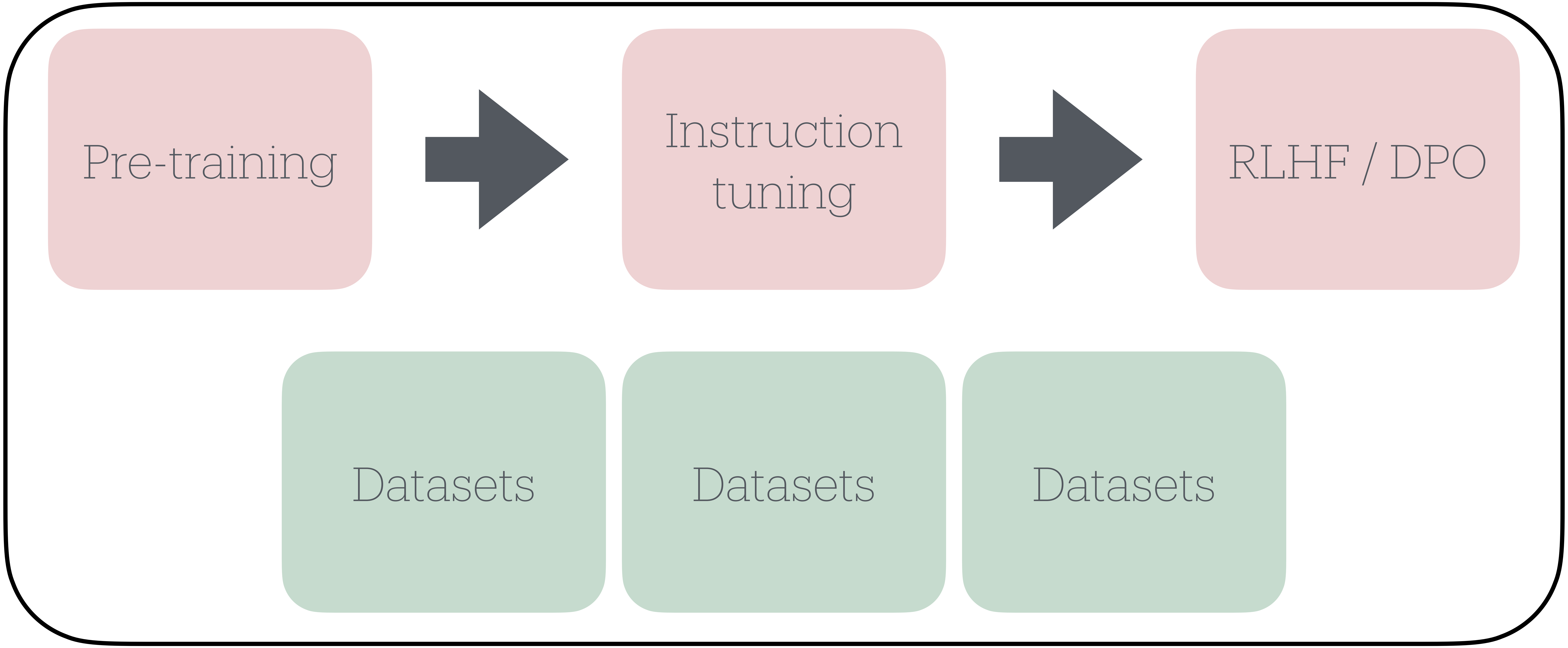
# Structured Dialogues

Philipp Krähenbühl, UT Austin

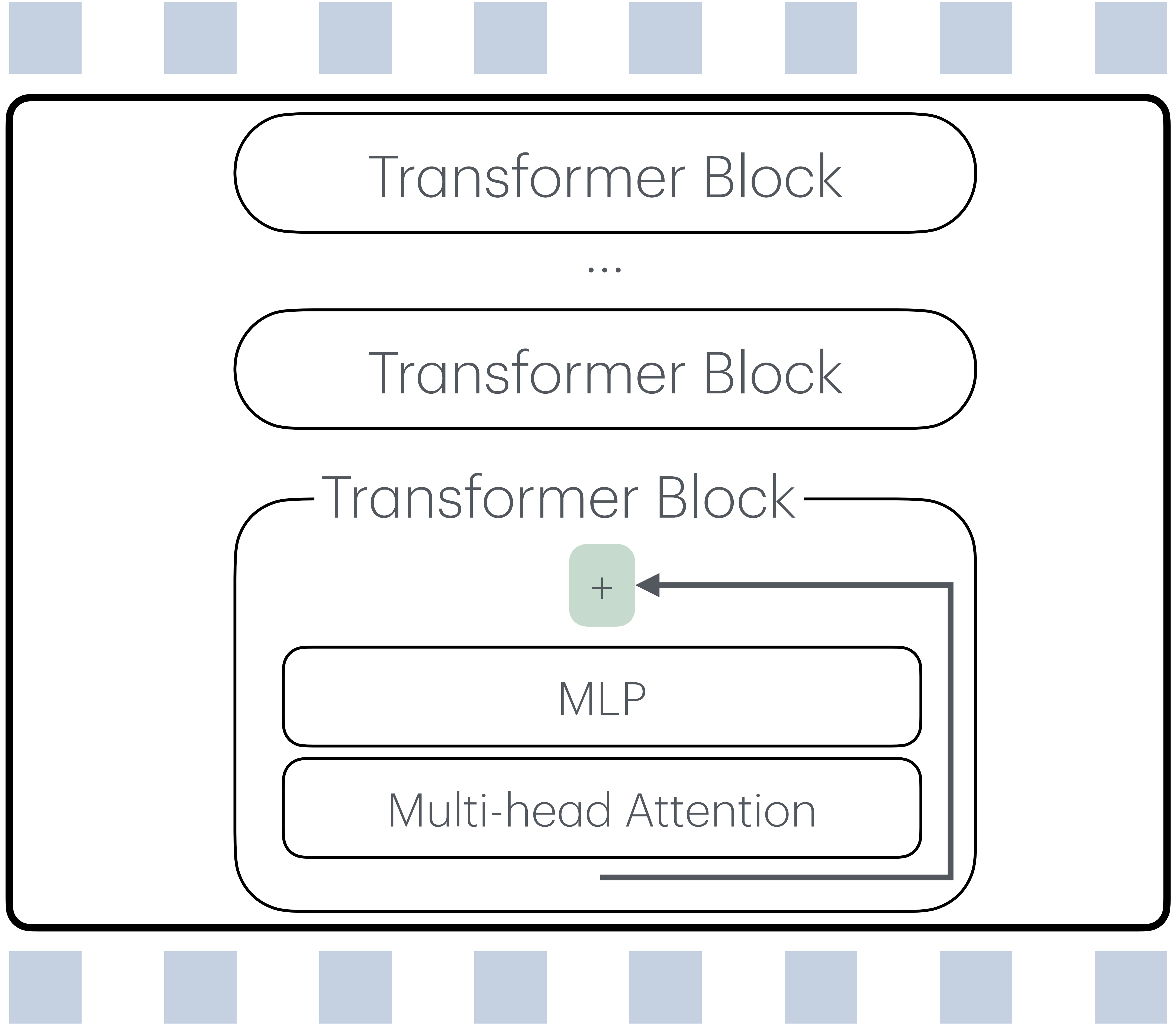
# Full Picture

## Basic LLM



# Where does a LLM store information?

- Their weights
  - MLP and attention [1]
- Special tokens / activations [2,3]
  - Large activations or registers
- Their context

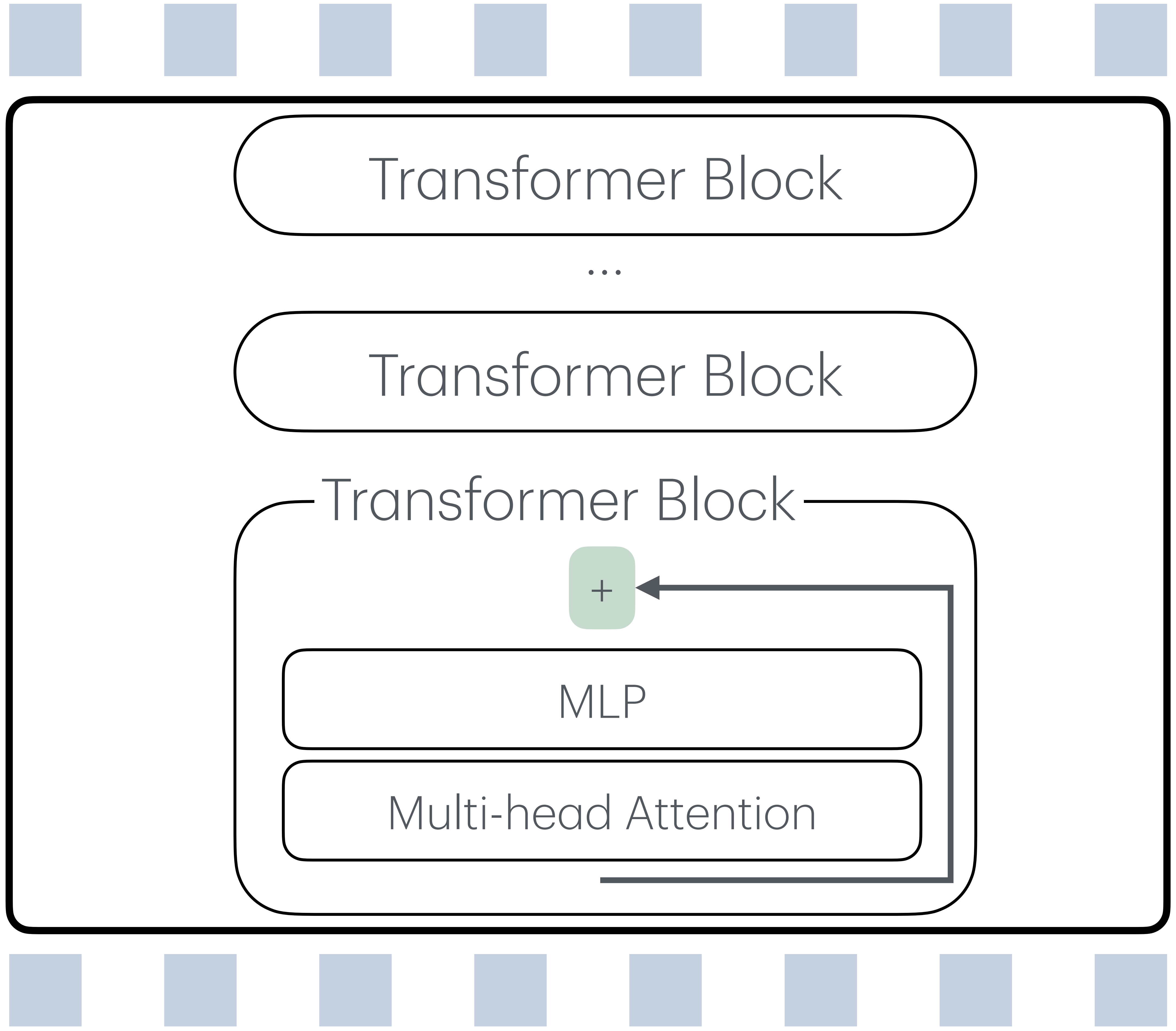




[1] Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws, Allen-Zhu 2024
[2] Vision Transformers Need Registers, Darcet etal 2023
[3] Massive Activations in Large Language Models, Sun etal 2024

# Information in weights

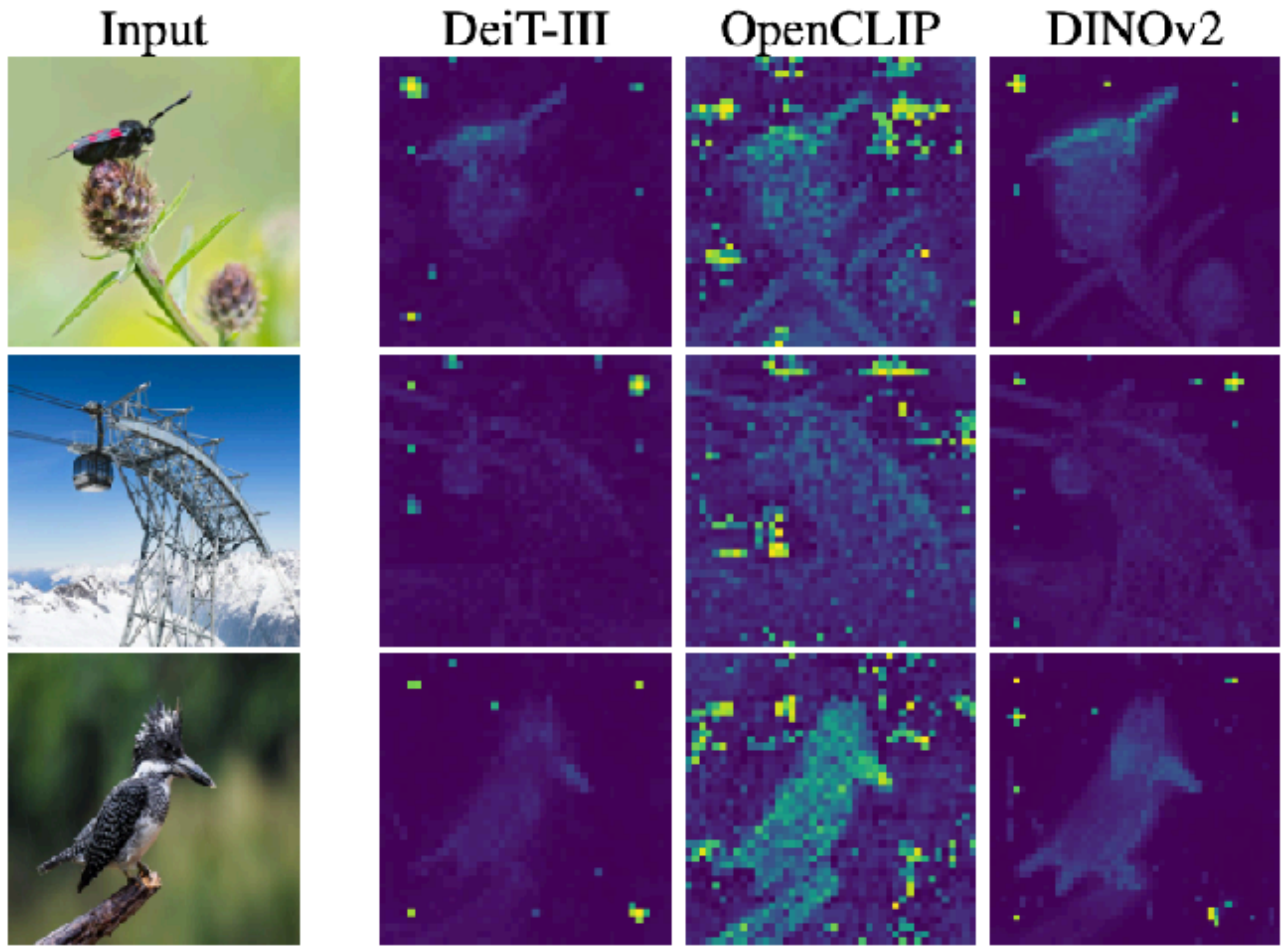
- LLMs can store up to 2 bits of information per weight [1]
  - In MLP
  - In Attention
  - 2 bits require very long training and multiple (up to 1000) augmentations of same information

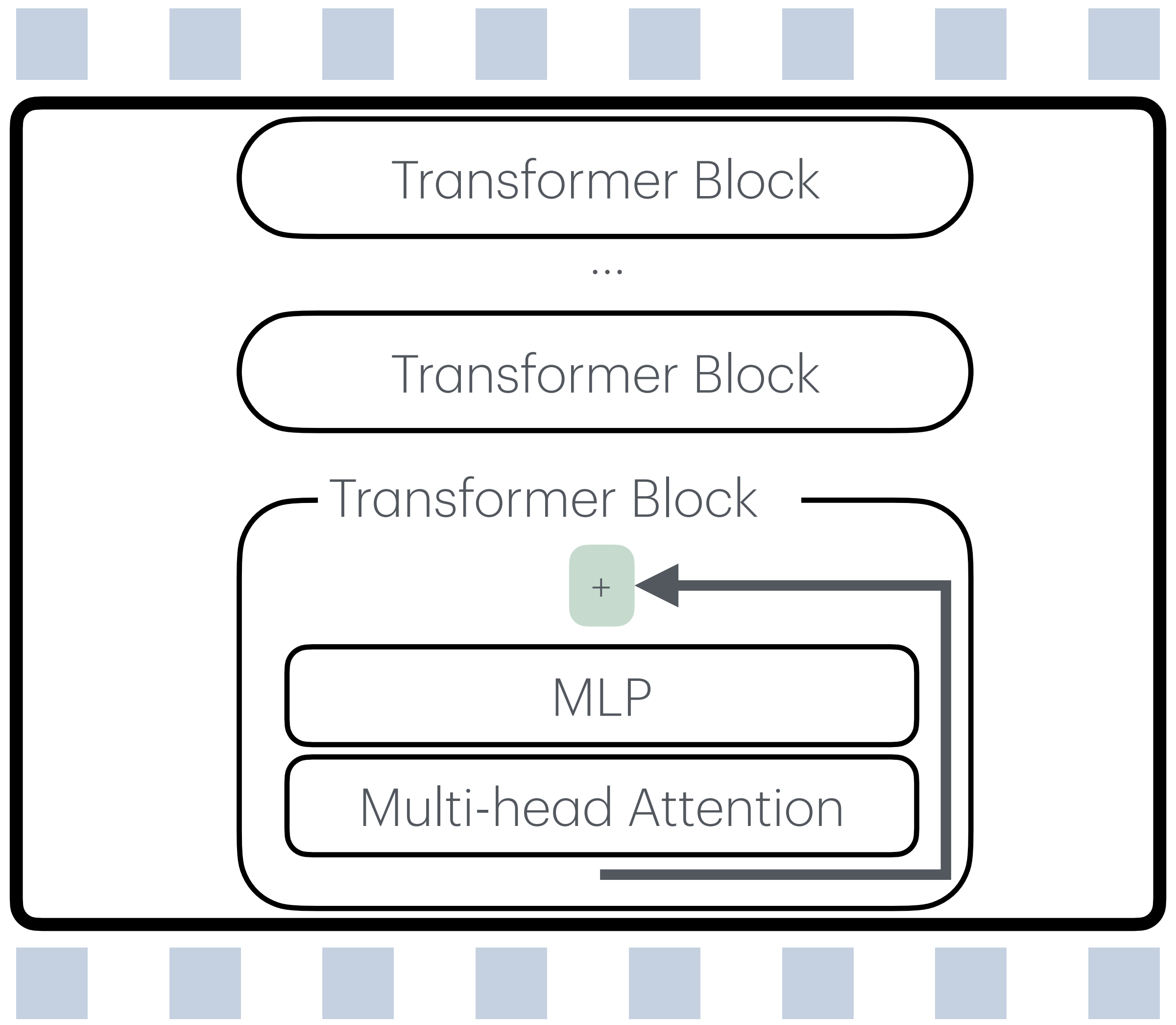



Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws, Allen-Zhu 2024

# Special tokens / activations

- LLMs use special tokens to store information
  - LLMs attend to <BOS> token
  - VLMs attend to background

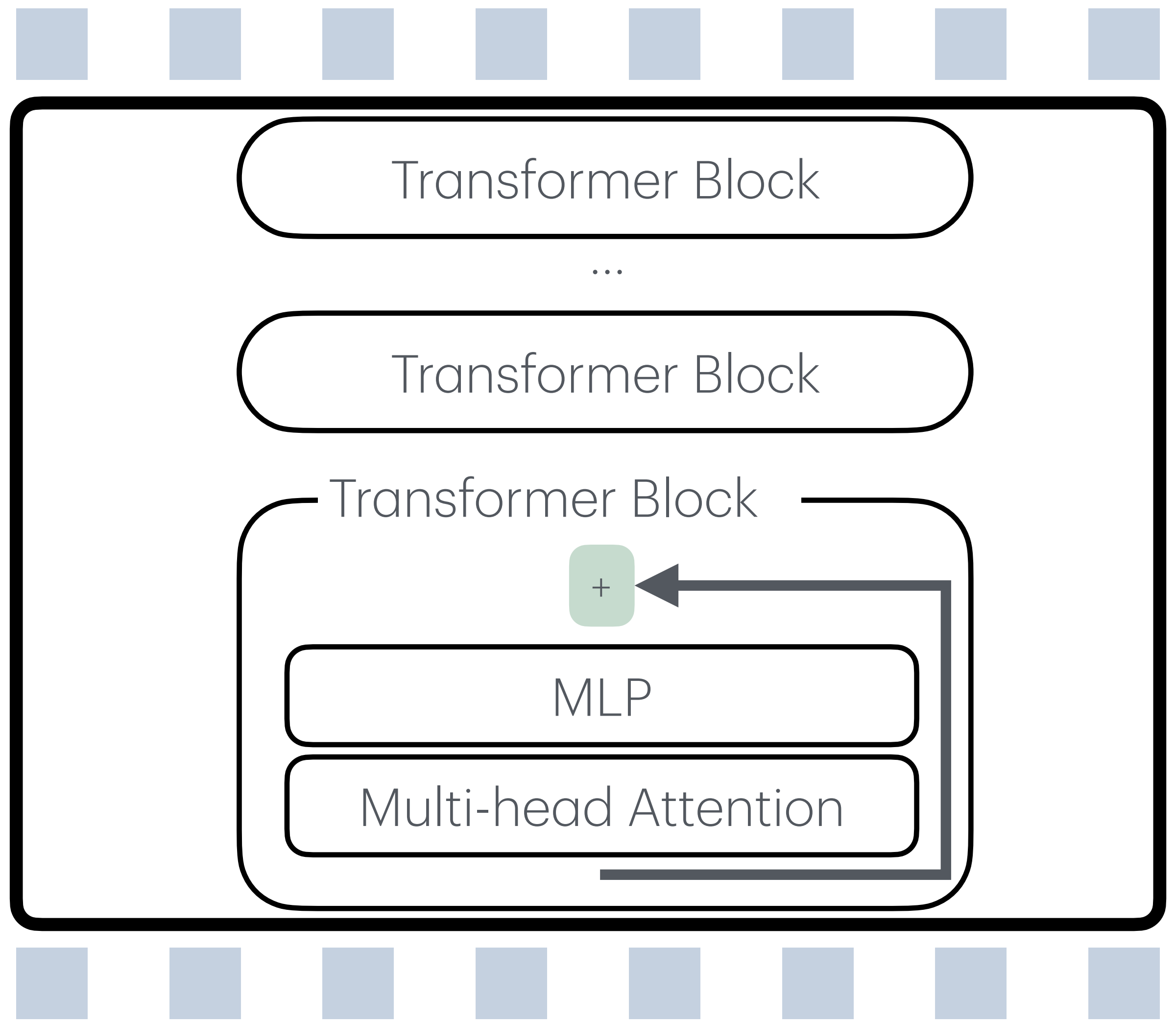



[1] Vision Transformers Need Registers, Darcet etal 2023
[2] Massive Activations in Large Language Models, Sun etal 2024

# Context

- LLMs store information in their context
- Examples
  - System prompt
  - Retrieval Augmented Generation
  - …




[1] Vision Transformers Need Registers, Darcet etal 2023
[2] Massive Activations in Large Language Models, Sun etal 2024

# In context learning

- Describe the task
  - Give examples input - output pairs
  - Then ask for your specific

```
Translate words from English to German using
JSON as an output. Here are some examples

Car
{“English”: “Car”, “German”: “Auto”}

Sun
{“English”: “Sun”, “German”: “Sonne”}

Moon
```

Language Models are Few-Shot Learners, Brown etal 2020

# In context learning

## Why does it work?

- LLMs like repeating patterns
  - Likely exist in pre-training data
- Examples of in-context prompts and answers during training (instruction tuning, alignment)

```
Translate words from English to German using
JSON as an output. Here are some examples

Car
{“English”: “Car”, “German”: “Auto”}

Sun
{“English”: “Sun”, “German”: “Sonne”}

Moon
```

Language Models are Few-Shot Learners, Brown etal 2020

# In context learning

## What does it work for?

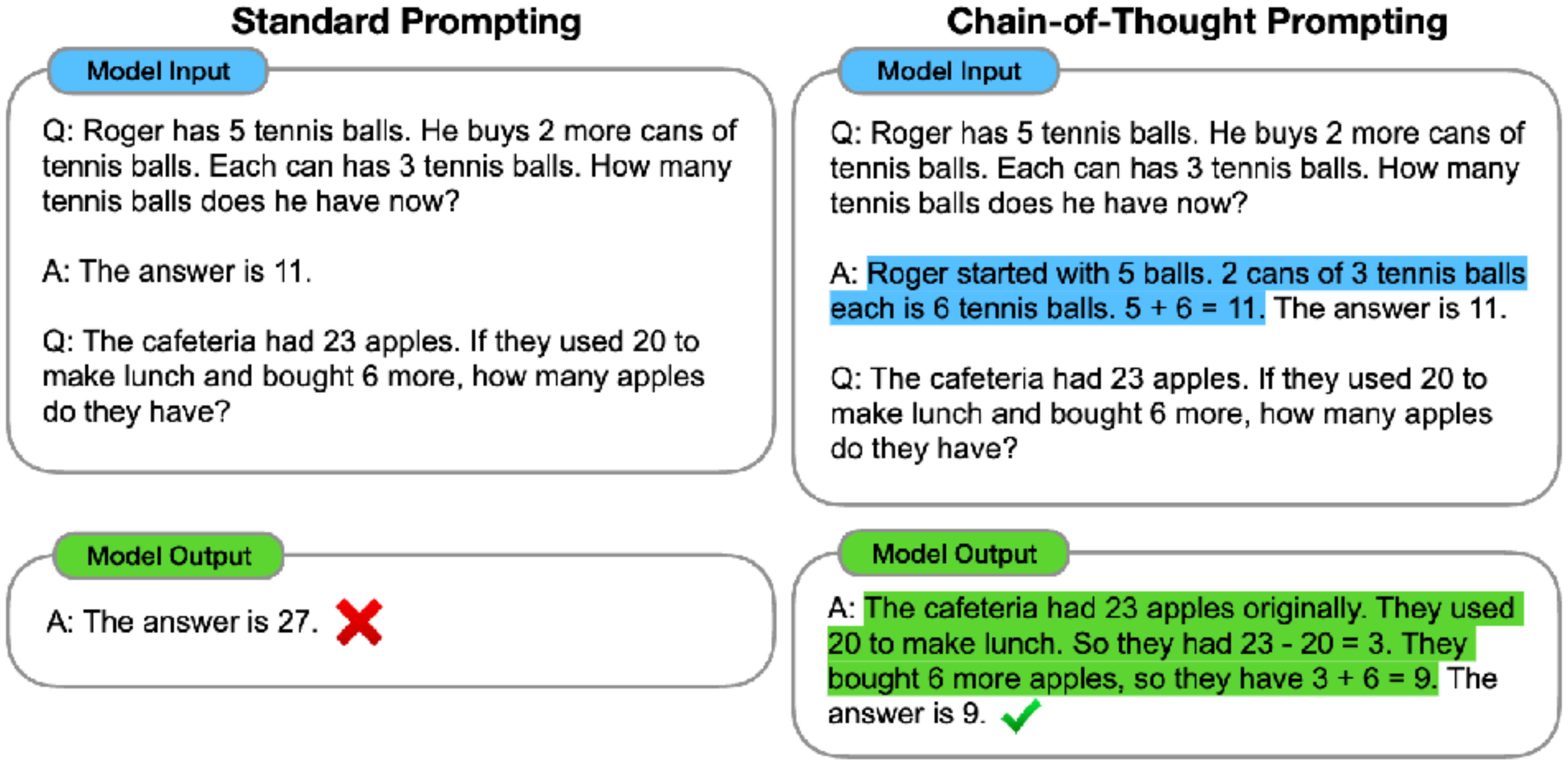
- Formatting outputs
- Simple requests

```
Translate words from English to German using
JSON as an output. Here are some examples

Car
{“English”: “Car”, “German”: “Auto”}

Sun
{“English”: “Sun”, “German”: “Sonne”}

Moon
```

Language Models are Few-Shot Learners, Brown etal 2020

# Chain of thought

- Ask model to derive answer
  - Pre-instruction tuning: In-context example of reasoning
  - Post-instruction tuning
    - Ask model to think step-by-step before giving the answer
    - Guide model through thinking process

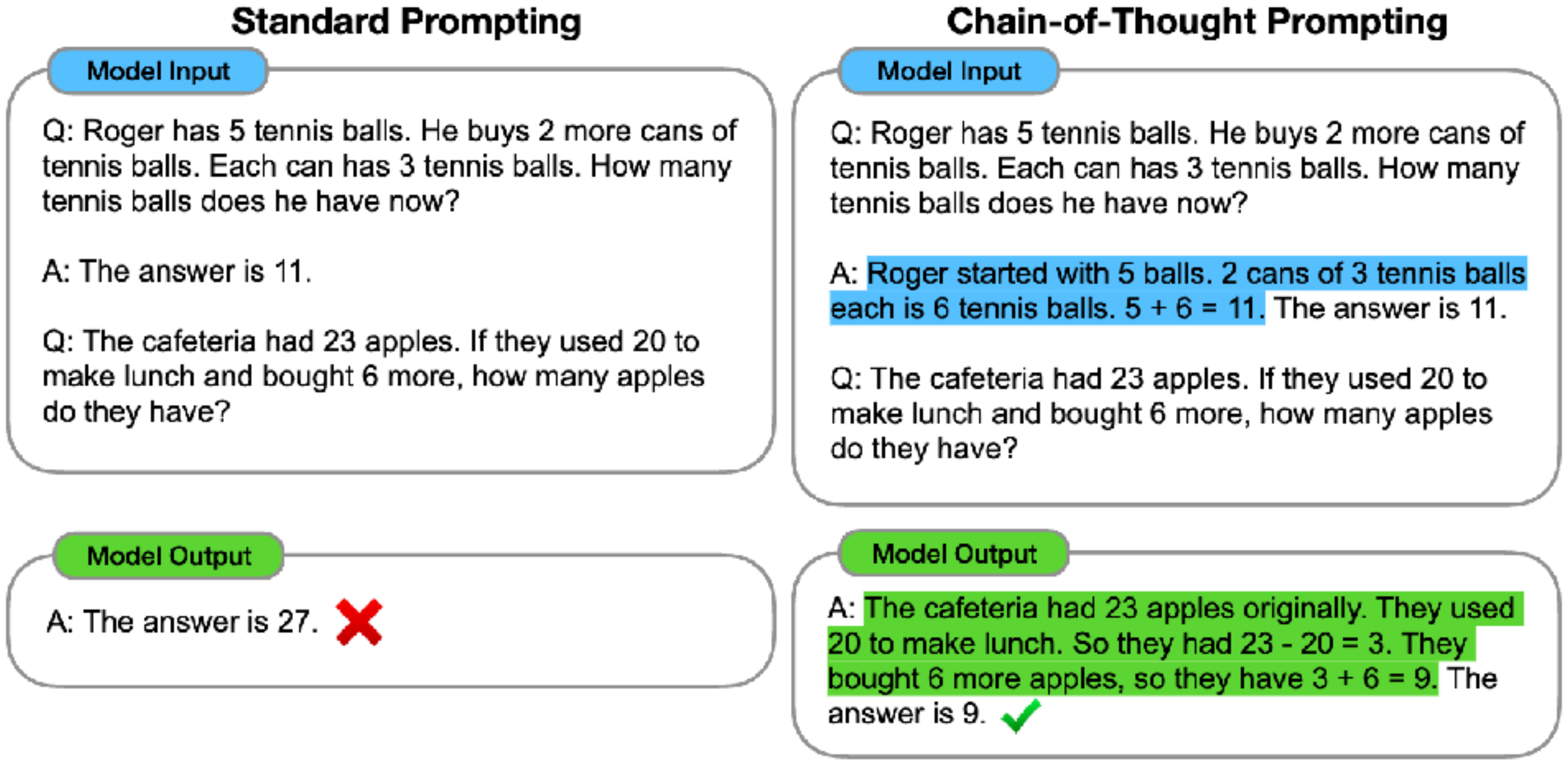



Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, Wei etal 2022

# Chain of thought

## Why does it work?

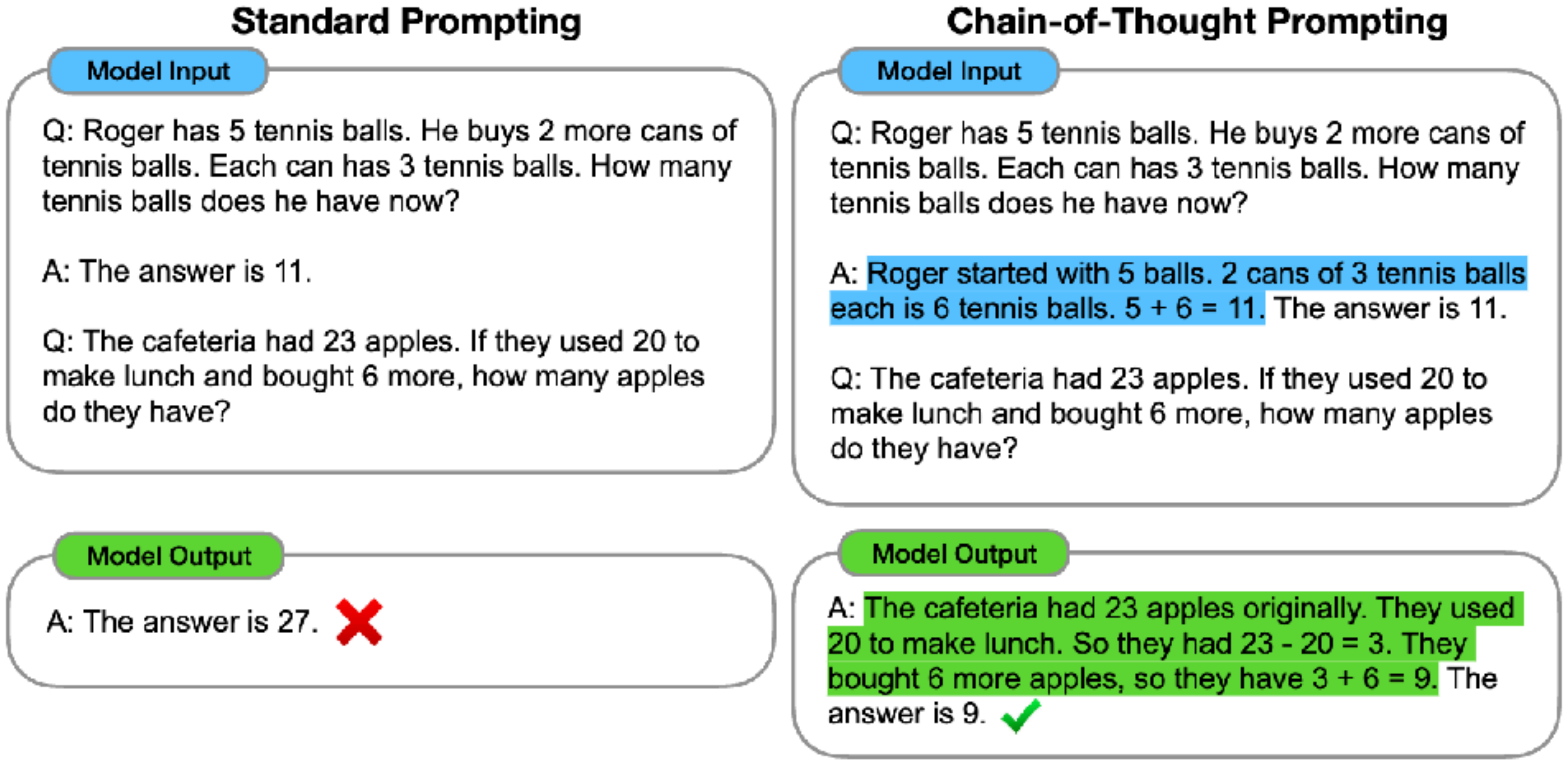
- More output tokens = better performance
    - Delays making a decision
- Can work around tokenization issues
    - Break up numbers




Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, Wei etal 2022

# Chain of thought

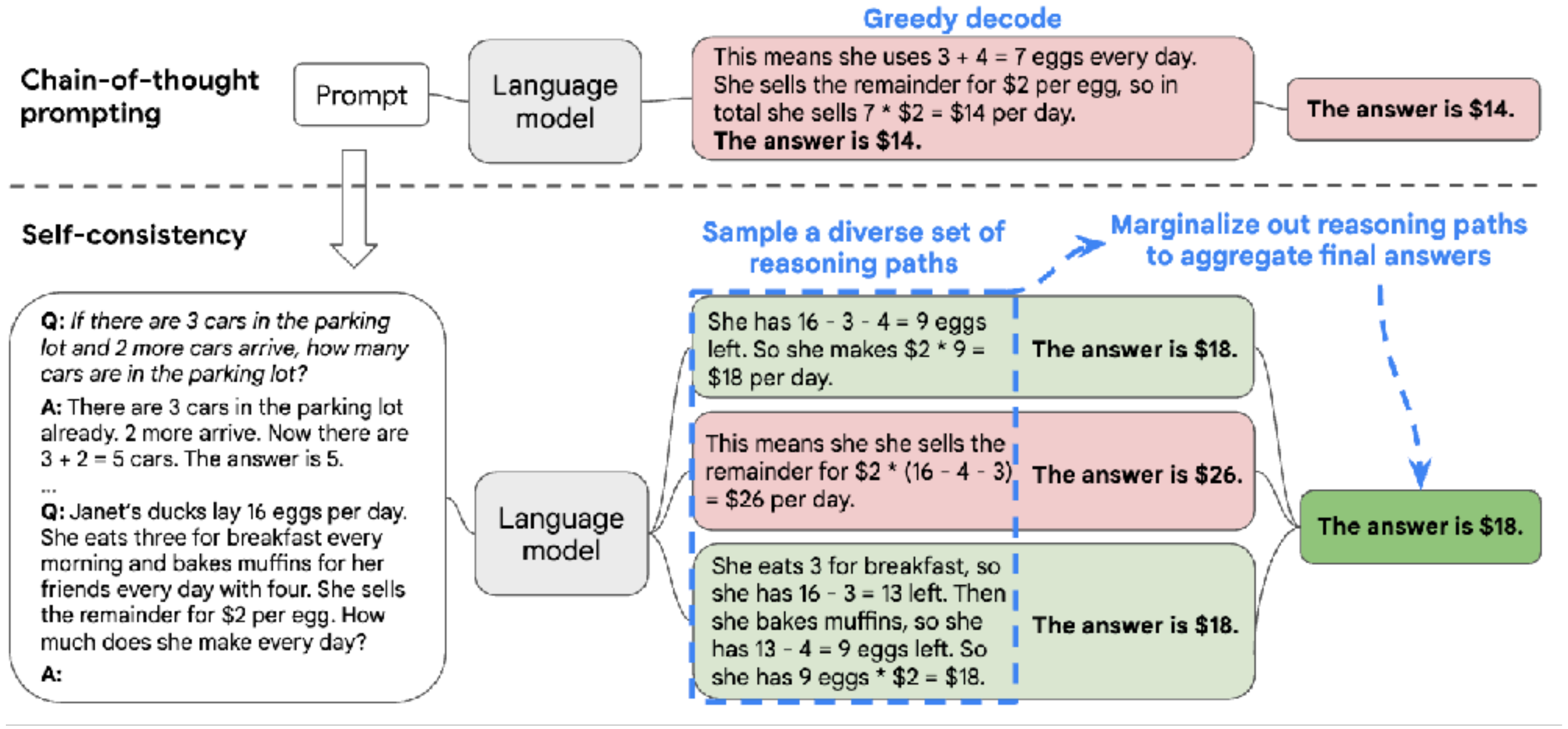
- Order matters
  - Think first, then answer
  - Chain-of-BS: Ask model to give answer and justify it




Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, Wei etal 2022

# Self-Consistency

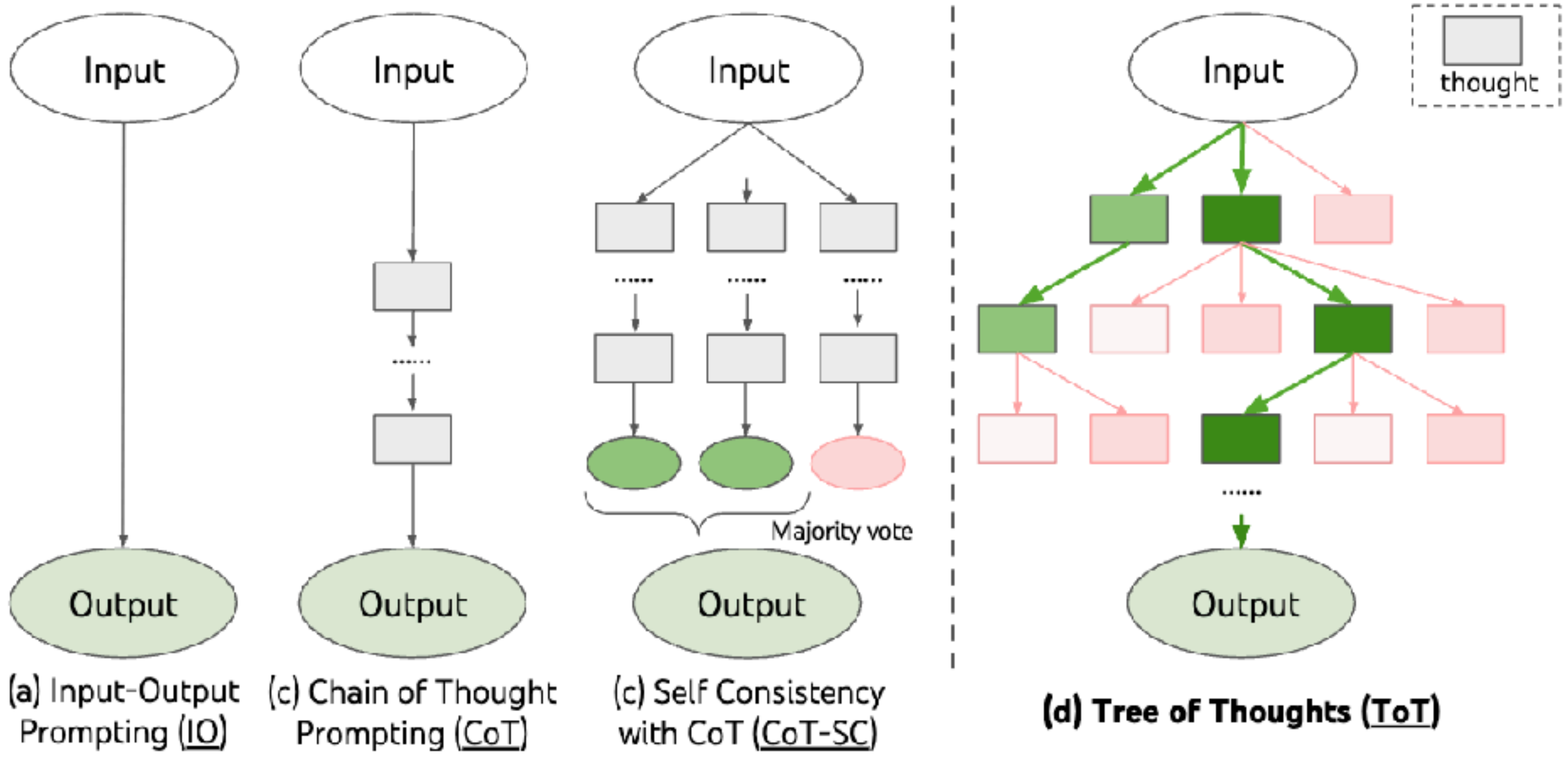
- Let the model reason multiple times
  - Pick the most frequent answer
  - Mathematically: Marginalize out reasoning to obtain most likely answer

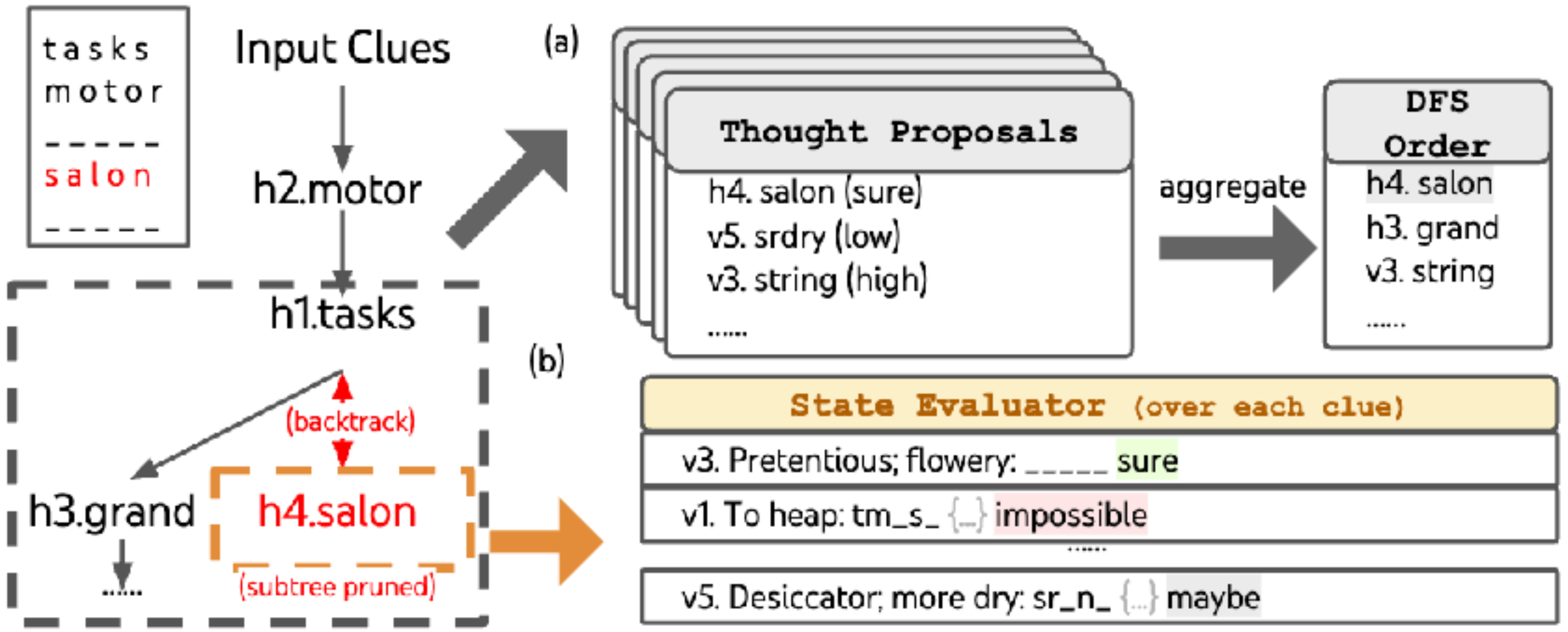



Self-Consistency Improves Chain of Thought Reasoning in Language Models, Wang etal 2022

# Tree of Thoughts

- Combine tree search with CoT
  - Requires a state-evaluator (i.e. reward/cost/scoring function or second LLM)

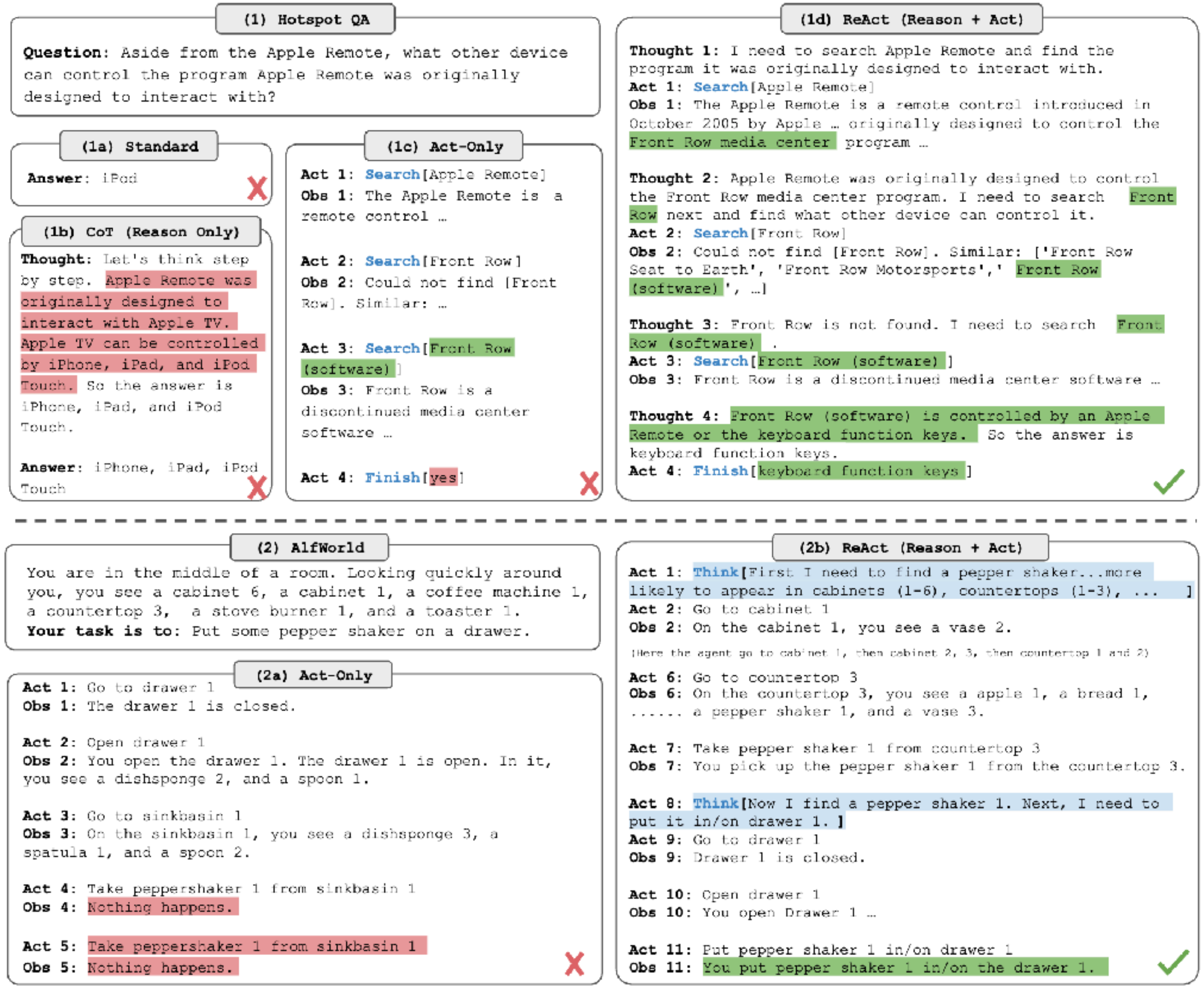



Tree of Thoughts: Deliberate Problem Solving with Large Language Models, Yao etal 2023

# ReACT

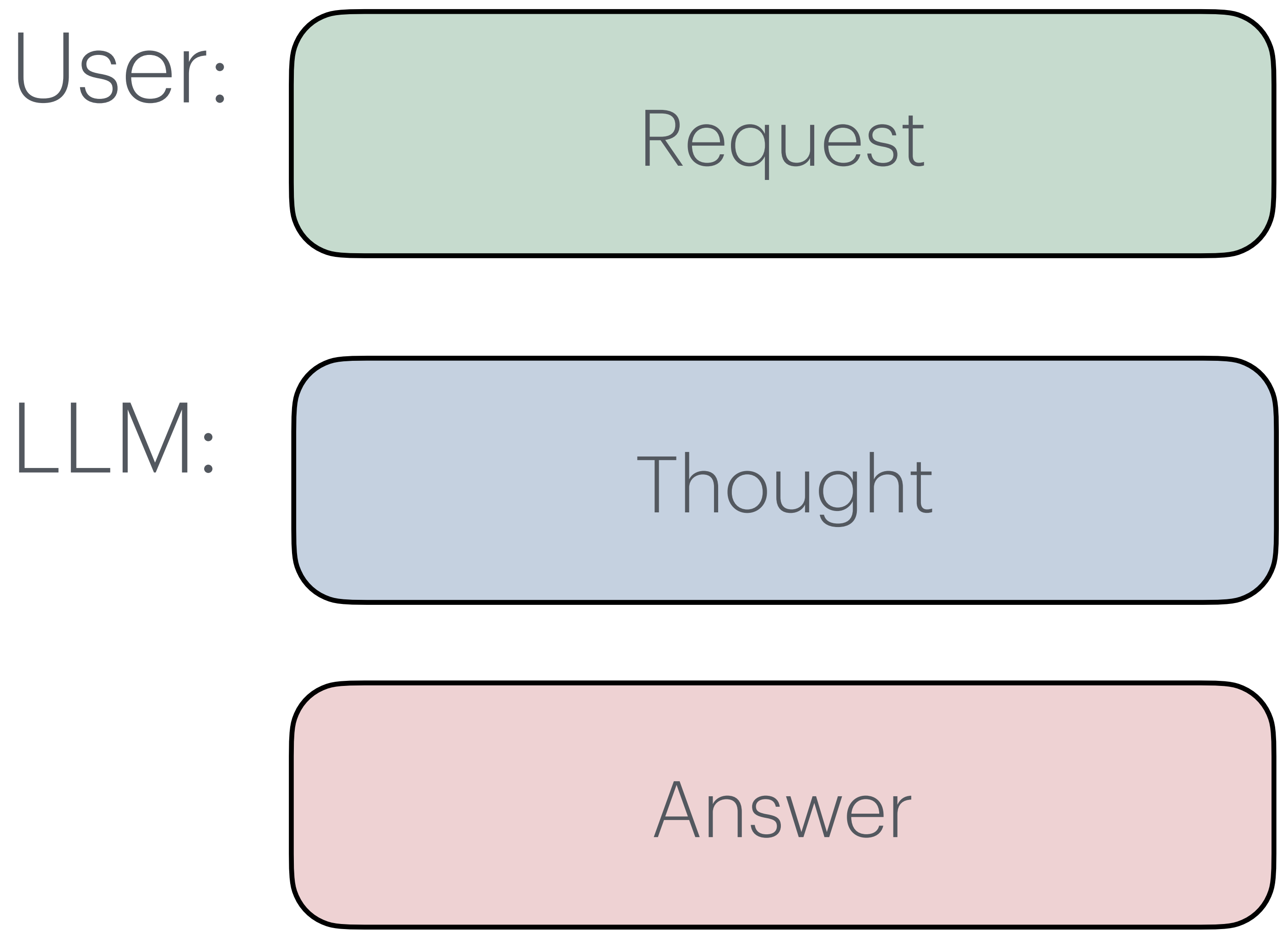
- Chain of thought for iterative actions / tool use
  - Thought
  - Action
  - Observation (from external tool)



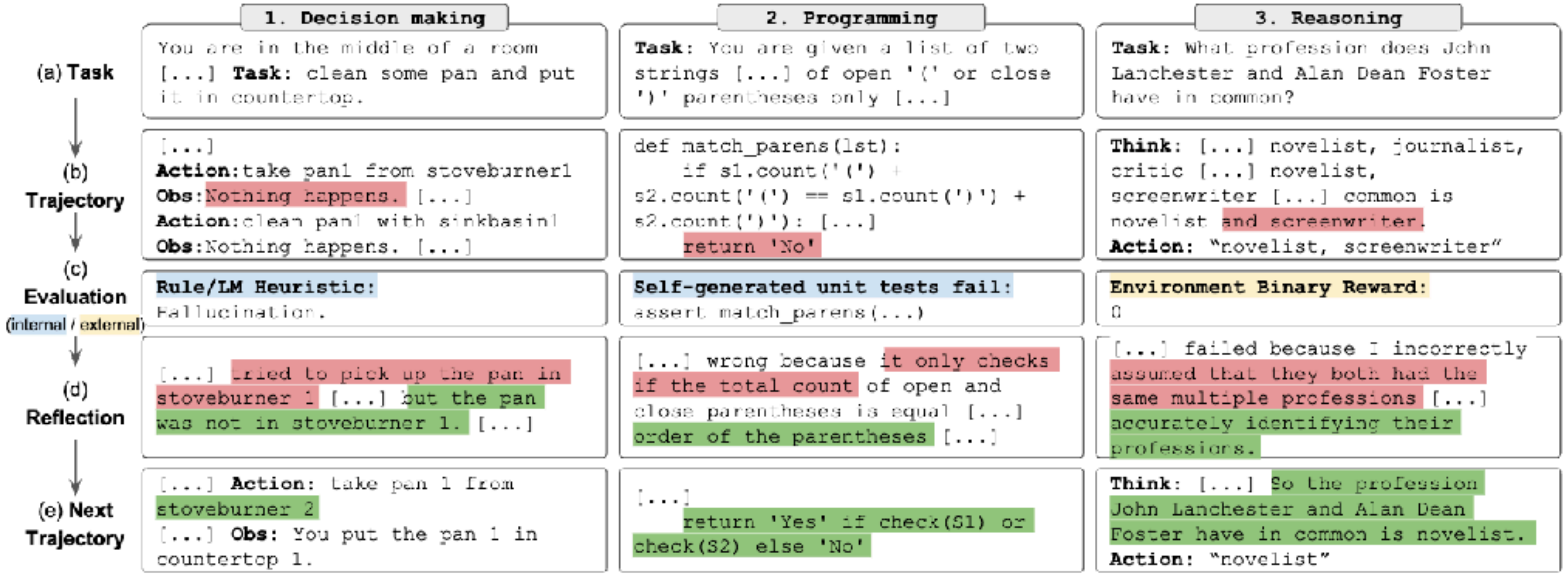

ReAct: Synergizing Reasoning and Acting in Language Models, Yao etal 2022

# Structured Dialogues

- Break down problem / tasks for LLM
  - Higher performance
  - Lots of human engineering / prompting



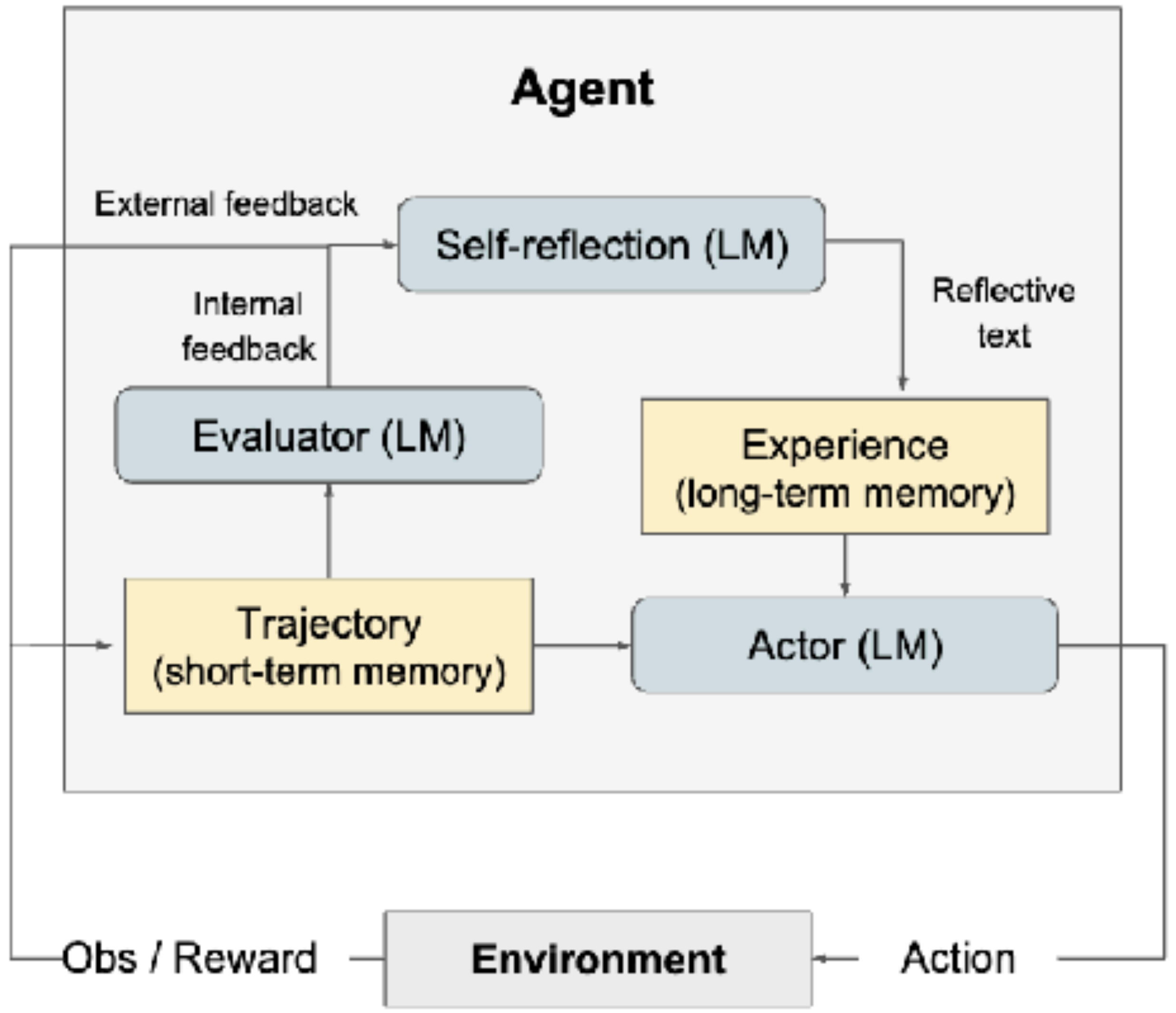
# Reflexion




- Chain of Thought / ReACT
- Obtain observation / result
- Reflect on outcome
- Repeat

Reflexion: Language Agents with Verbal Reinforcement Learning, Shin etal 2023

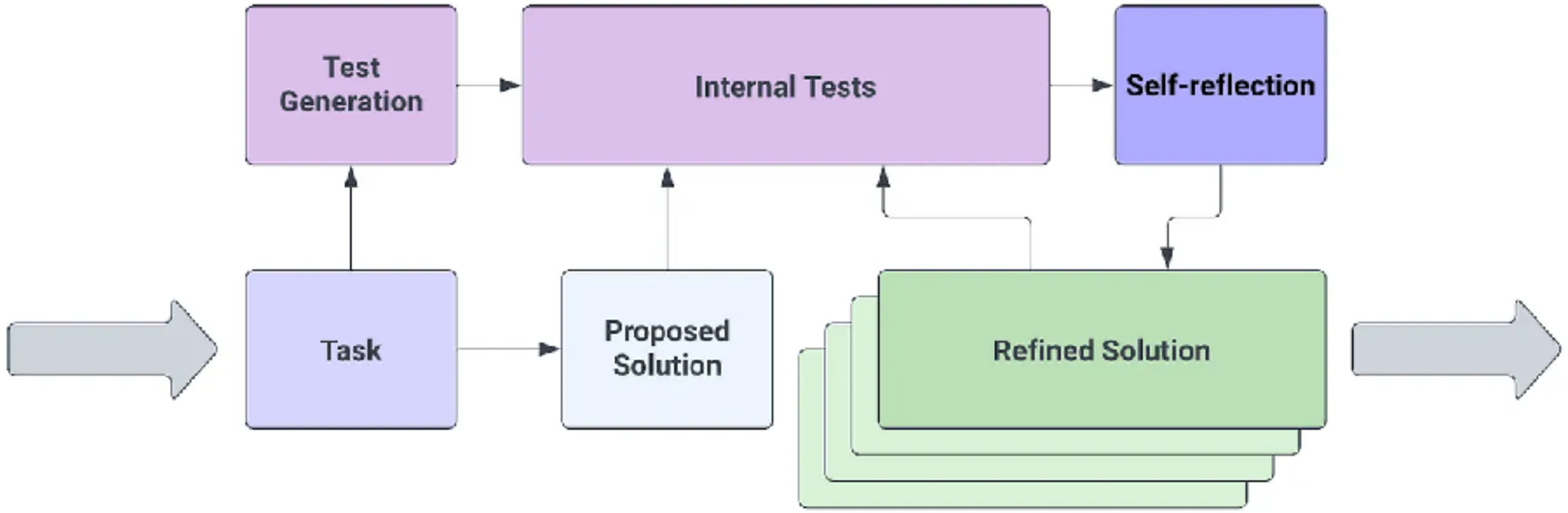
# Reflexion




**Algorithm 1** Reinforcement via self-reflection

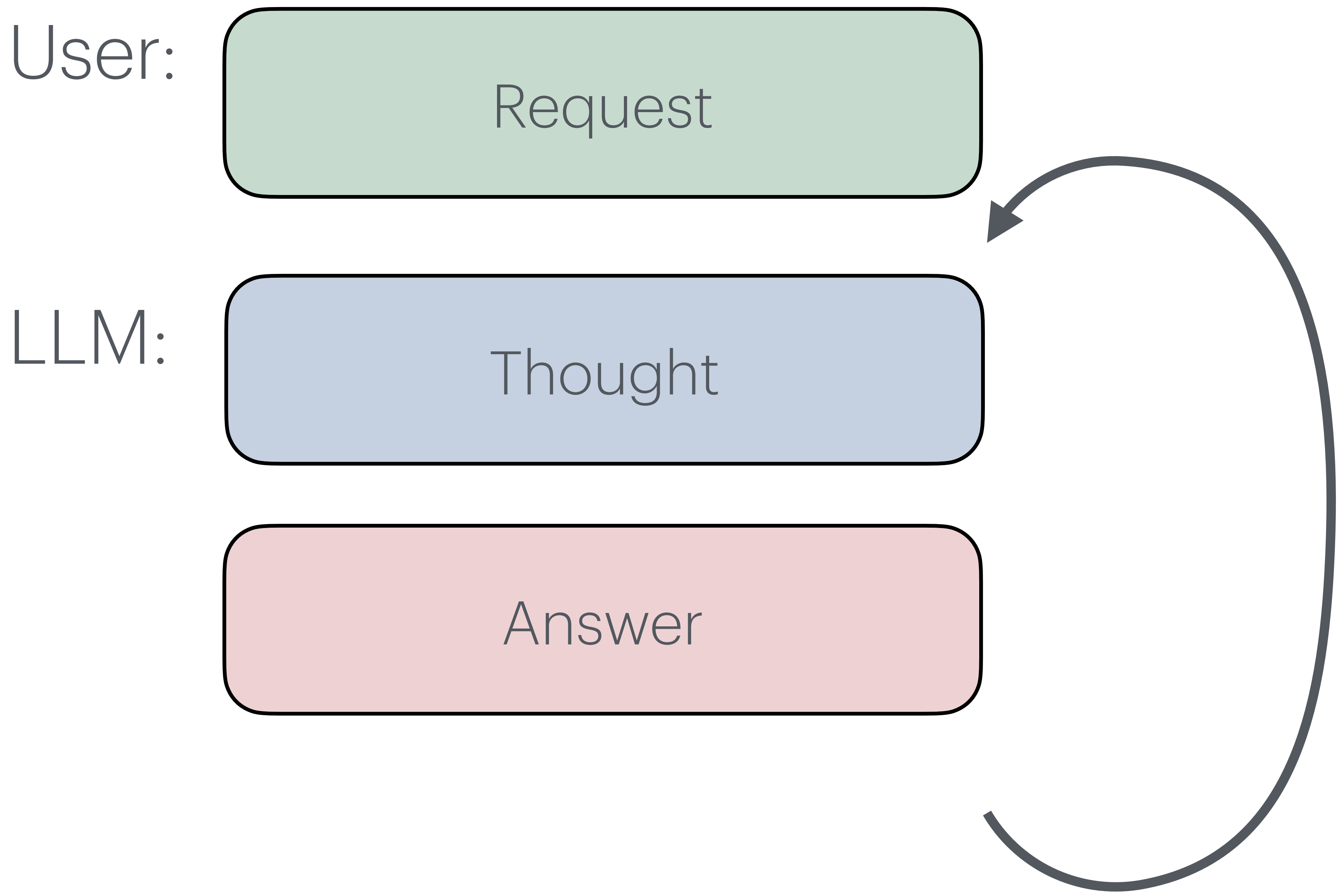
Initialize Actor, Evaluator, Self-Reflection:
$M_a$, $M_e$, $M_{sr}$
Initialize policy $\pi_\theta(a_i|s_i)$, $\theta = \{M_a, mem\}$
Generate initial trajectory using $\pi_\theta$
Evaluate $\tau_0$ using $M_e$
Generate initial self-reflection $sr_0$ using $M_{sr}$
Set $mem \leftarrow [sr_0]$
Set $t = 0$
**while** $M_e$ not pass or $t <$ max trials **do**
  Generate $\tau_t = [a_0, o_0, \ldots a_i, o_i]$ using $\pi_\theta$
  Evaluate $\tau_t$ using $M_e$
  Generate self-reflection $sr_t$ using $M_{sr}$
  Append $sr_t$ to $mem$
  Increment $t$
**end while**
**return**

- Connections to reinforcement learning
  - More strictly planning
- Requires a evaluator (cost function)
  - External environment (i.e. simulator, code interpreter)
  - LLM generated tests
  - Trained LLM verifier [1]




[1] Generative Verifiers: Reward Modeling as Next-Token Prediction, Zhang etal 2024

# Reflexion




- Break down problem / tasks for LLM
  - Higher performance
  - Lots of human engineering / prompting

# References


- [1] Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws, Allen-Zhu 2024
- [2] Vision Transformers Need Registers, Darcet etal 2023
- [3] Massive Activations in Large Language Models, Sun etal 2024
- [4] Language Models are Few-Shot Learners, Brown etal 2020
- [5] Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, Wei etal 2022
- [6] Self-Consistency Improves Chain of Thought Reasoning in Language Models, Wang etal 2022
- [7] Tree of Thoughts: Deliberate Problem Solving with Large Language Models, Yao etal 2023
- [8] ReAct: Synergizing Reasoning and Acting in Language Models, Yao etal 2022
- [9] Reflexion: Language Agents with Verbal Reinforcement Learning, Shin etal 2023
- [10] Generative Verifiers: Reward Modeling as Next-Token Prediction, Zhang etal 2024