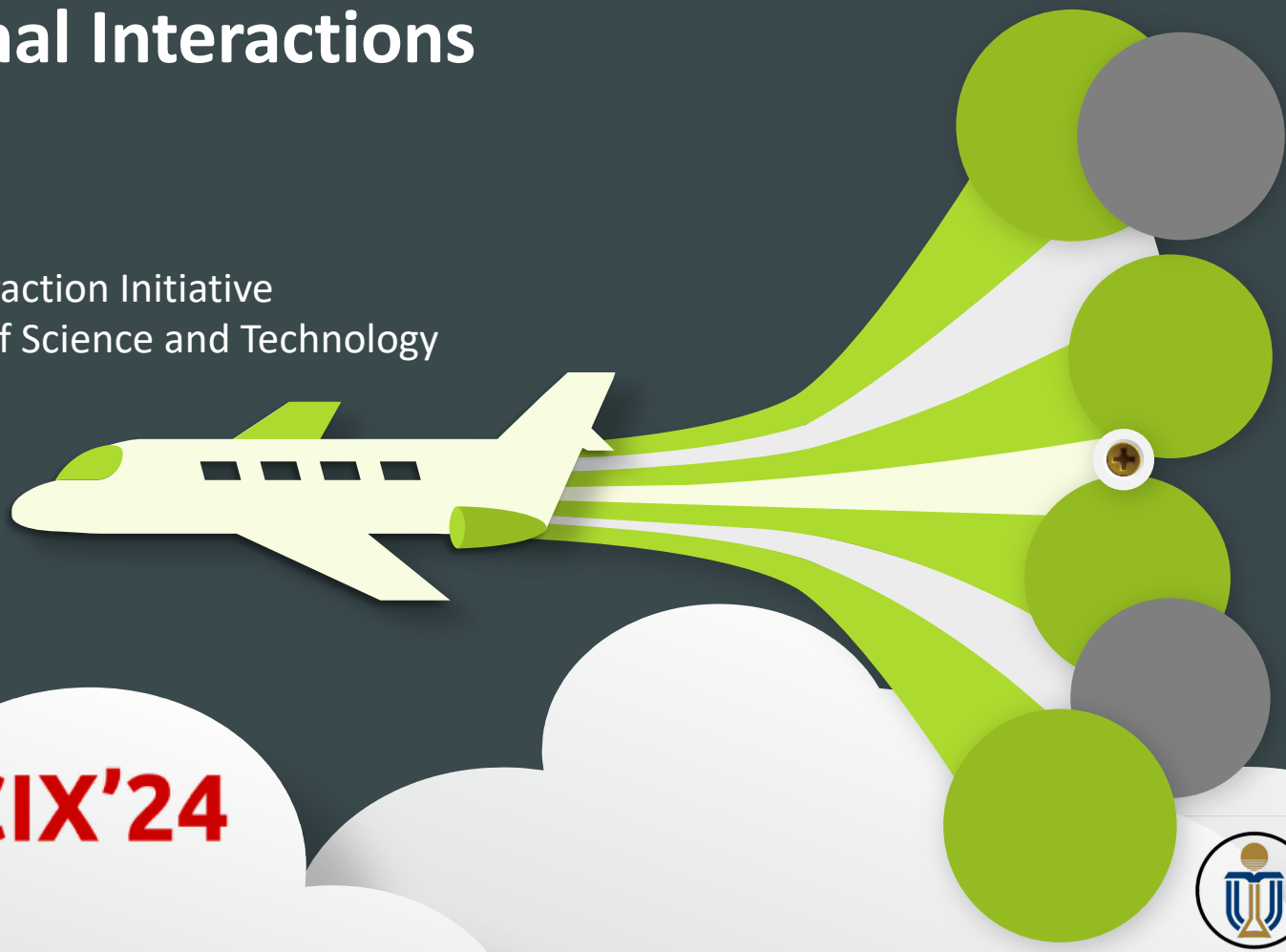
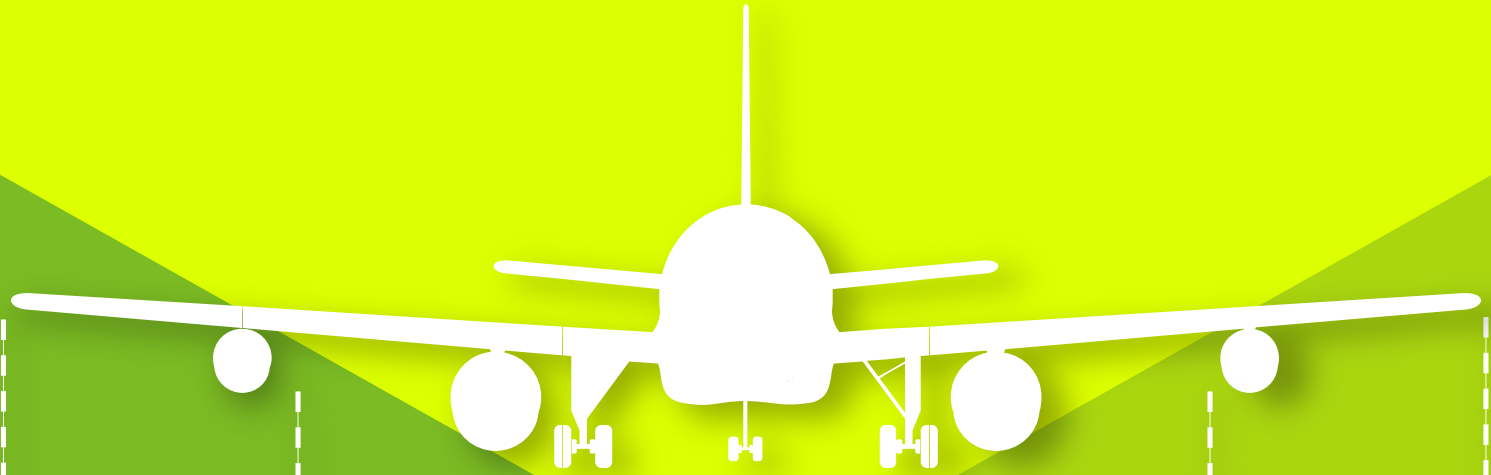
Charting the Routes of Thoughts in LLM-Empowered Conversational Interactions

Xiaojuan Ma

Human-Computer Interaction Initiative
Hong Kong University of Science and Technology
mxj@cse.ust.hk



CIX'24



01 Introduction

02 Chain-of-Thought

03 Tree-of-Thoughts

04 Graph-of-Thoughts

05 Applications

06 Exercise

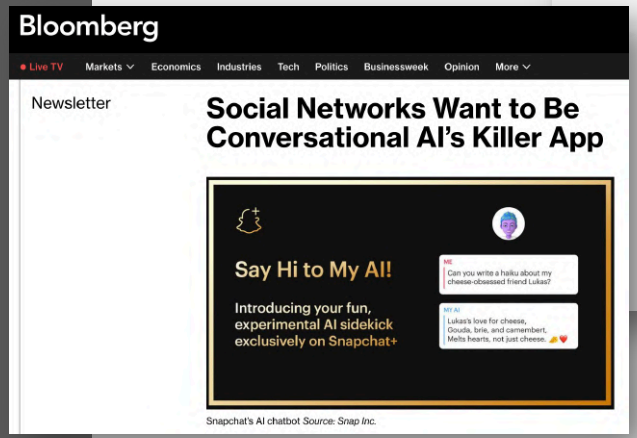
01 Introduction

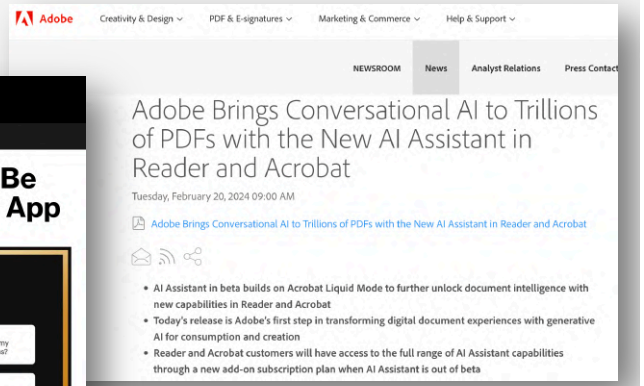
Affordable and Scalable Services

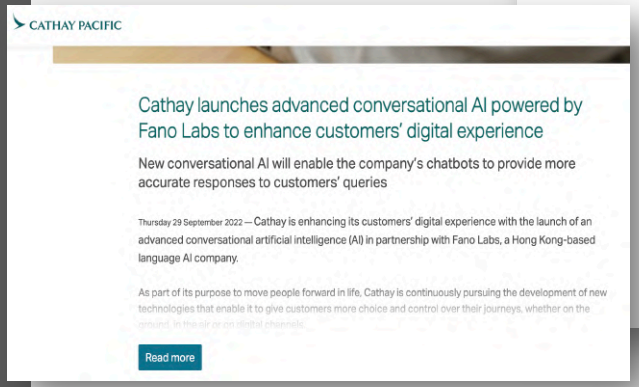
Natural Interactions

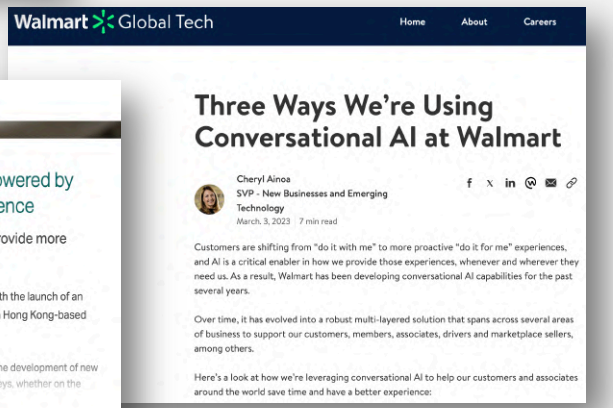
Personalization

Fast Development of Foundation Models (e.g., LLMs, VLMs, etc.)







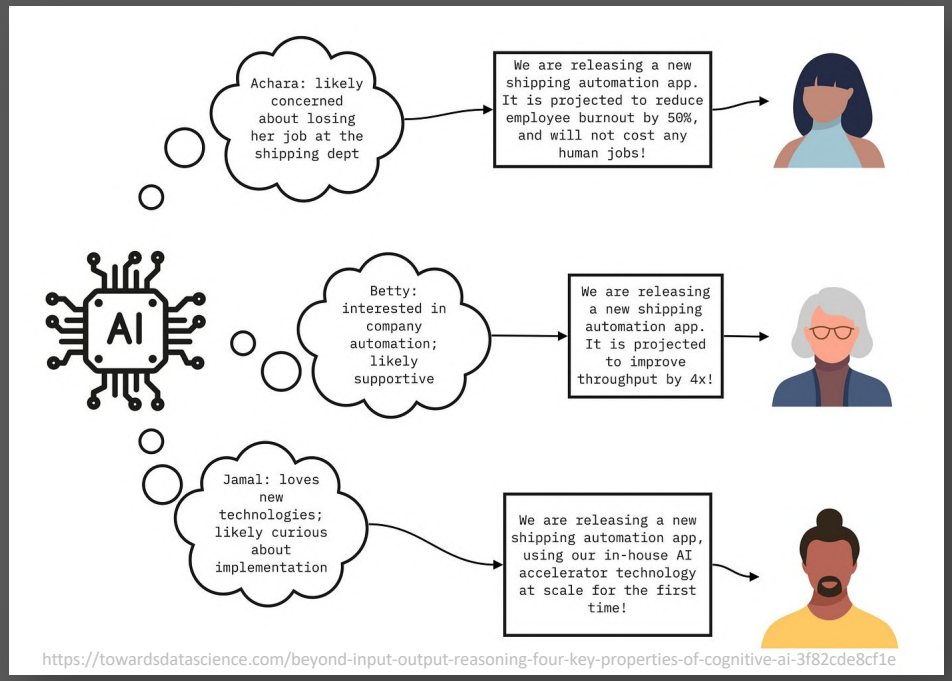


01 Introduction

Affordable and Scalable Services

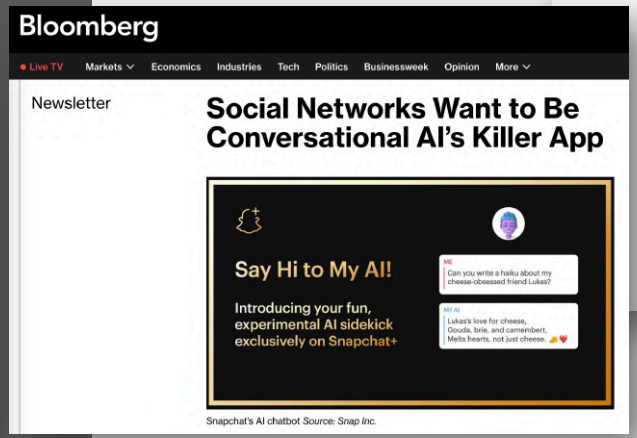
Natural Interactions

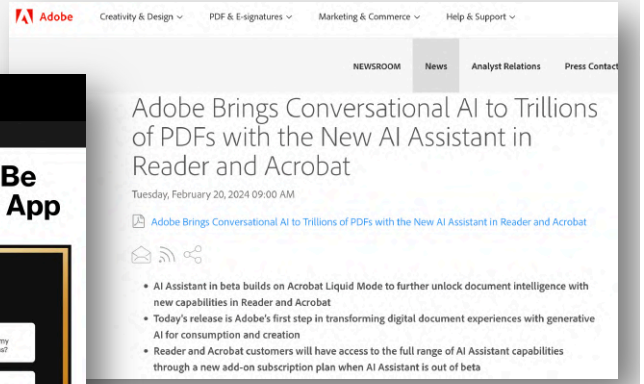
Personalization

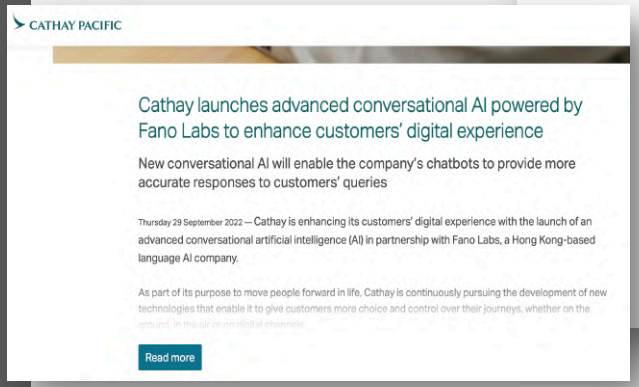


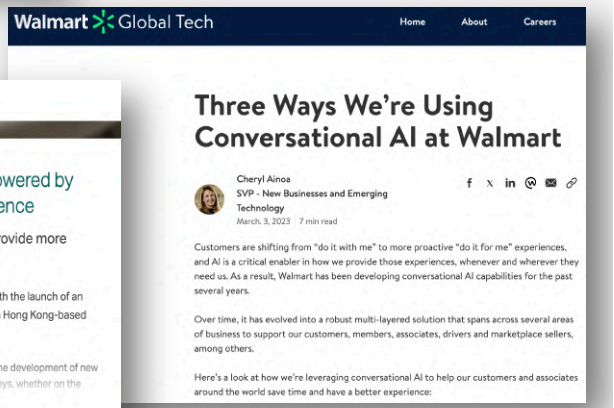
https://towardsdatascience.com/beyond-input-output-reasoning-four-key-properties-of-cognitive-ai-3f82cde8cf1e

Fast Development of Foundation Models (e.g., LLMs, VLMs, etc.)





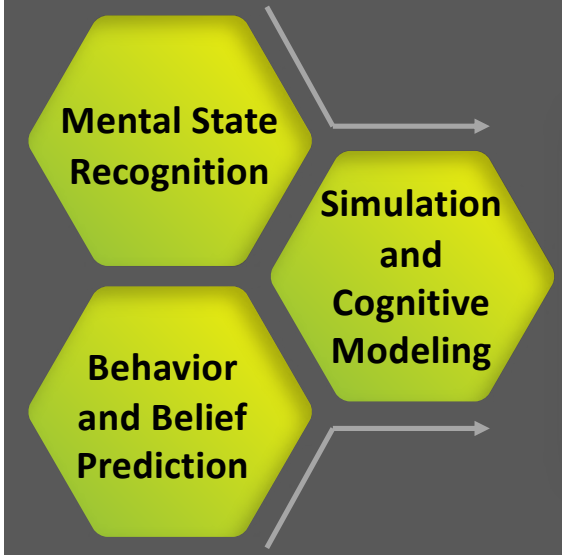


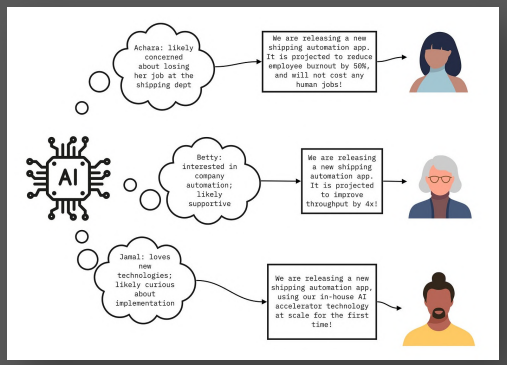


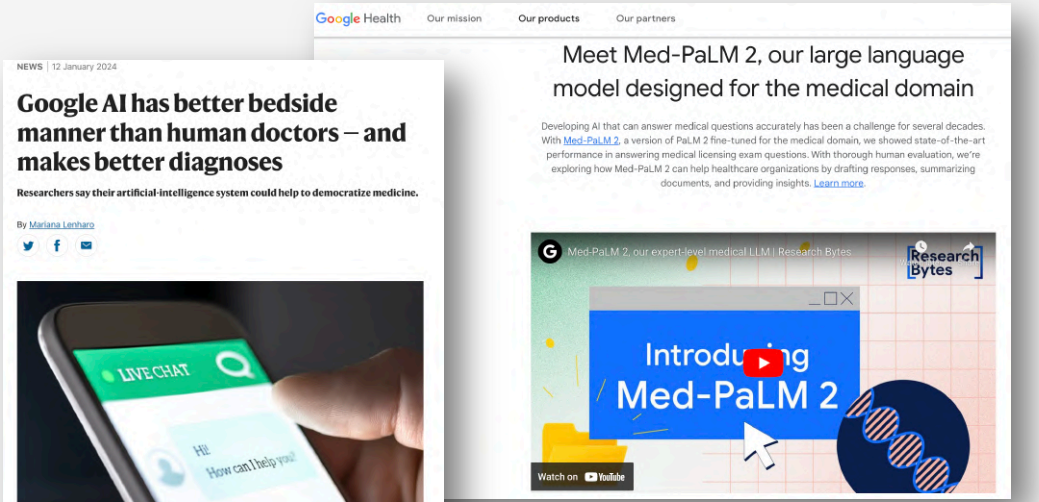
01 Introduction

Theory of Mind

Premack and Woodruff (1978) coined the concept and define **Theory of Mind (ToM)** as the ability that "… one infers states that are not directly observable and one uses these states anticipatorily, to predict the behavior of others as well as one's own." (p. 525)

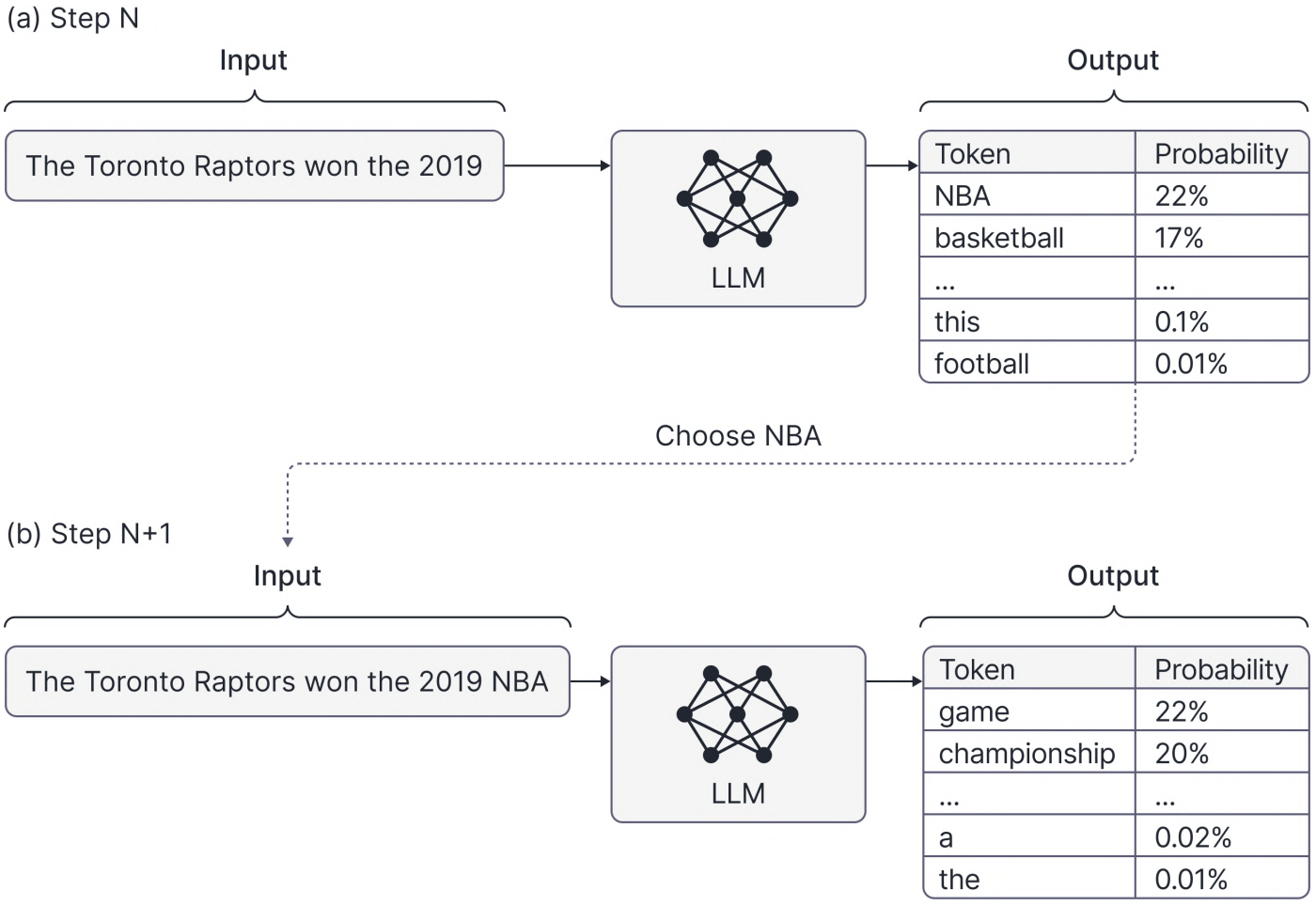






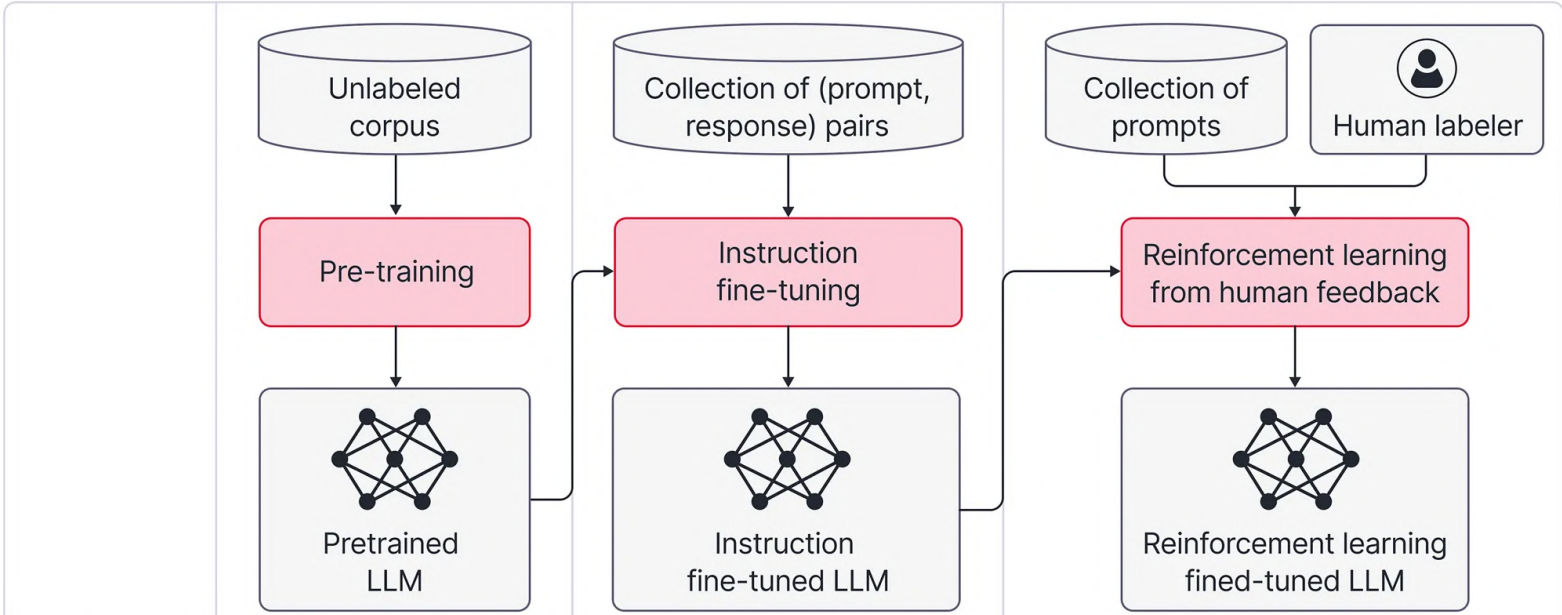
- Has current conversational AI (LLM-empowered) achieved ToM?
- How to equip conversational AI (LLM-empowered) with ToM?

Overview of a General LLM Processing Pipeline



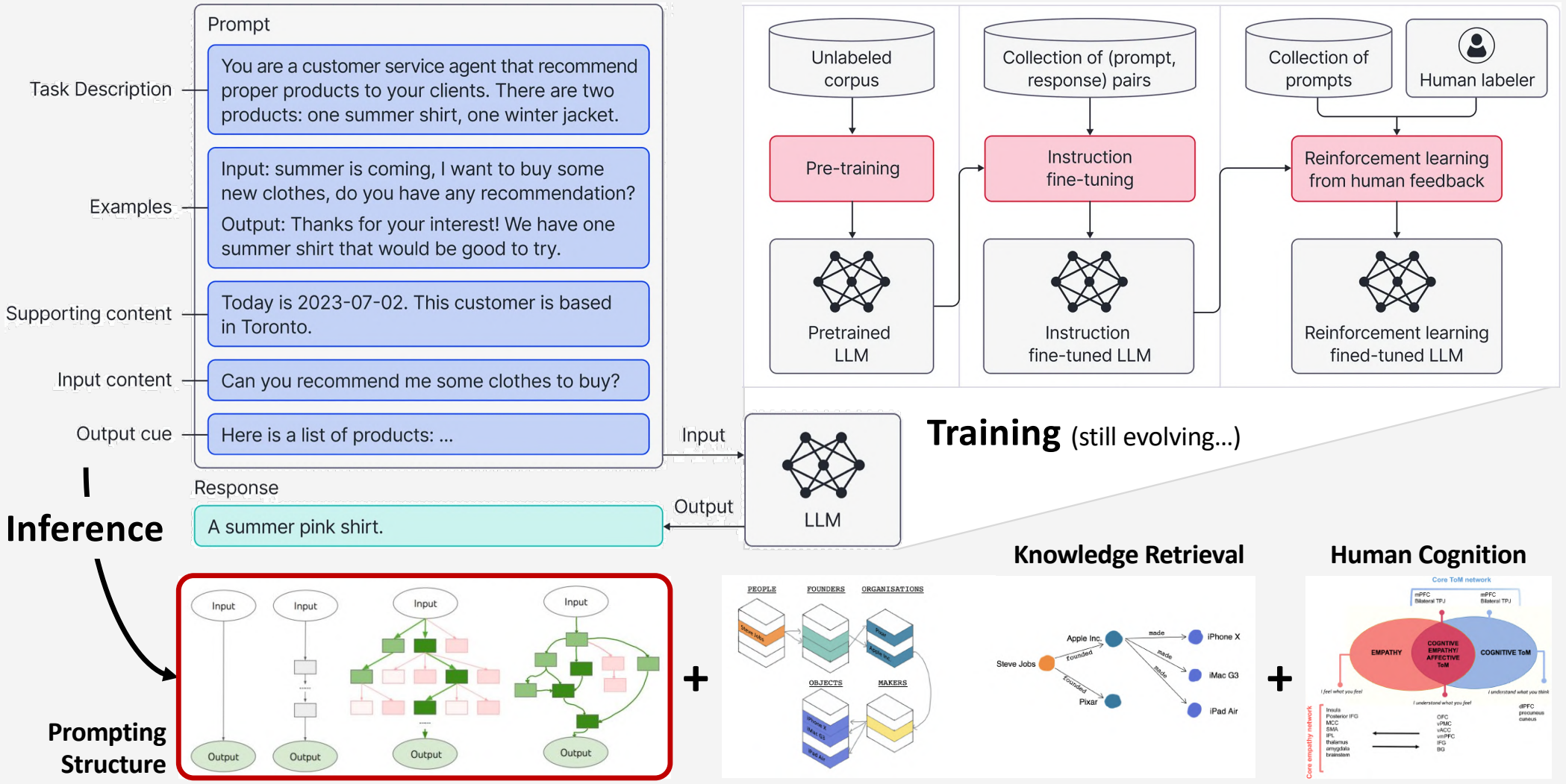
https://www.borealisai.com/research-blogs/a-high-level-overview-of-large-language-models/

Overview of a General LLM Processing Pipeline



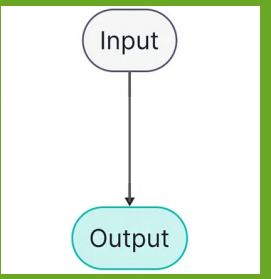
Dataset size	x00 billions to 1.x trillion tokens	~xk to x0k (prompt, response)	~x0k prompt
Example of models	GPT-3, LLaMA, Falcon, BLOOM	Dolly-v2, Falcon-instruct	Claude, GPT-4, ChatGPT
	(a)Pre-training	(b) Instruction fine-tuning	(c) Reinforcement learning from human feedback

https://www.borealisai.com/research-blogs/a-high-level-overview-of-large-language-models/



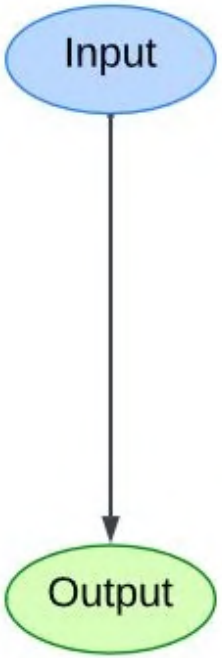
https://www.borealisai.com/research-blogs/a-high-level-overview-of-large-language-models/
https://thesciencemuseum.github.io/heritageconnector/post/2020/11/06/knowledge-graphs-machine-learning-and-heritage-collections/
https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3
https://www.researchgate.net/publication/356390737_Social_cognition_in_the_FTLD_spectrum_evidence_from_MRI/figures?lo=1&utm_source=google&utm_medium=organic

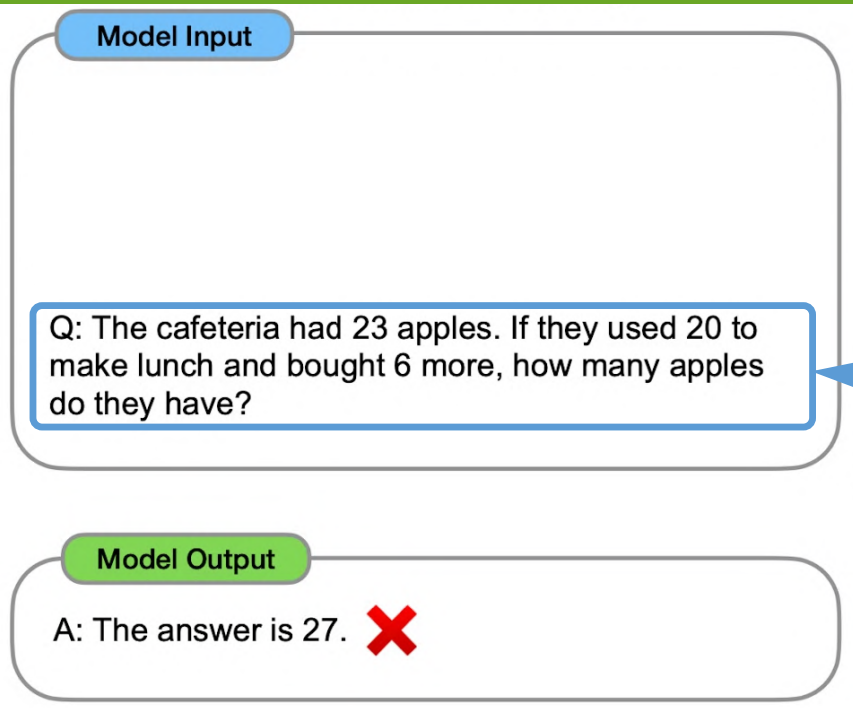
Prompt Engineering Techniques (Prompting Paradigms)



Zero-Shot

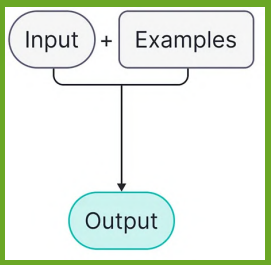
Standard Prompting





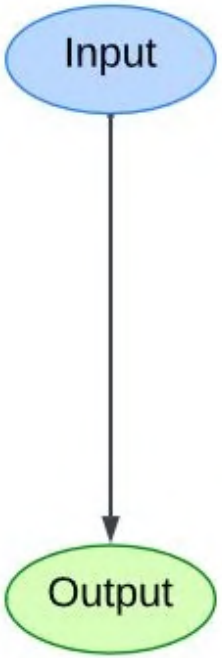
https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3
https://deepgram.com/learn/chain-of-thought-prompting-guide
https://arxiv.org/abs/2201.11903

Prompt Engineering Techniques (Prompting Paradigms)



Few-Shot

Standard Prompting



In-Context Example(s)

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?

Task Description, e.g., "Supposing you are …"

Supporting Content + Input Content

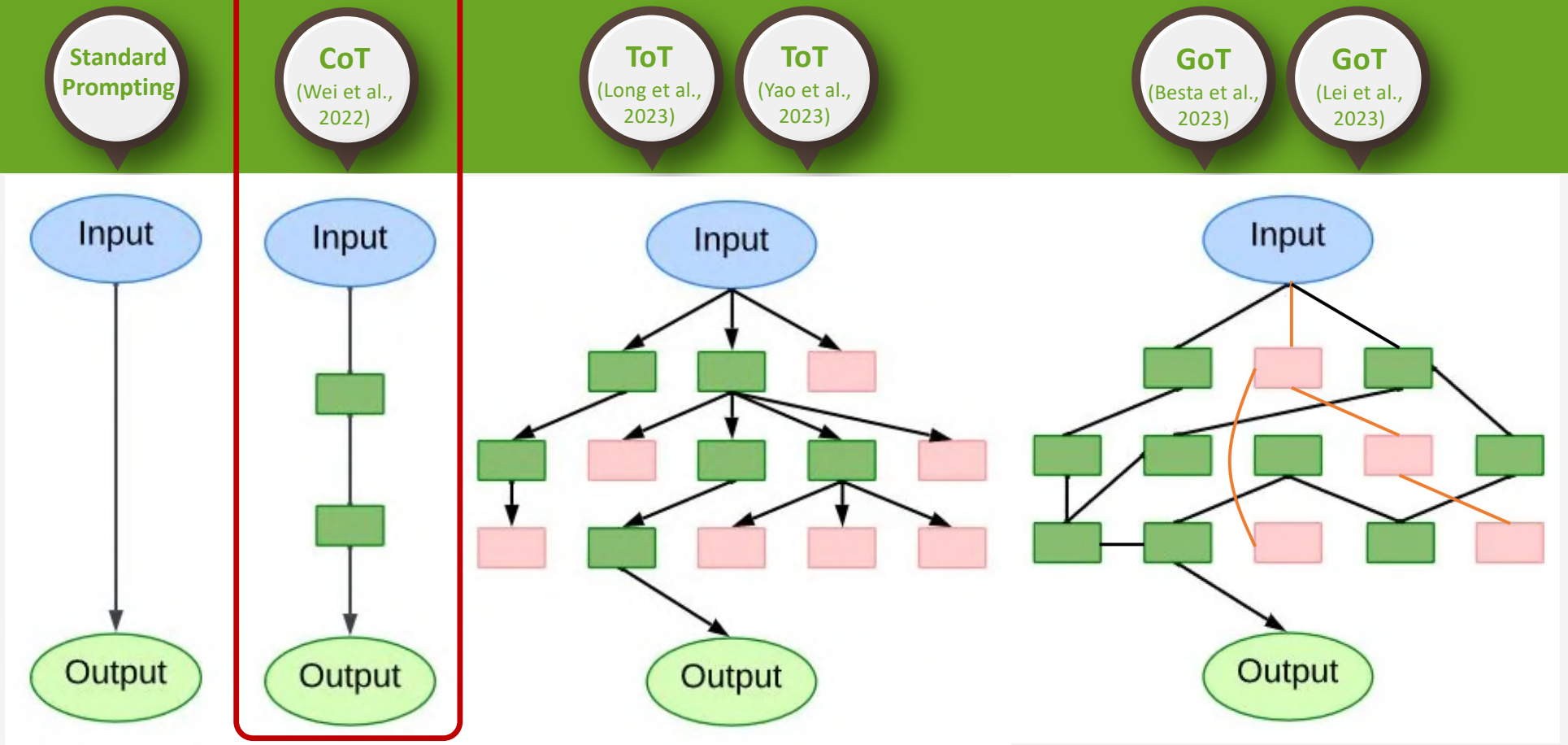
Output Cue(s), e.g., "Output in the format of...,"

Model Output

A: The answer is 27. ✗

https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3
https://deepgram.com/learn/chain-of-thought-prompting-guide
https://arxiv.org/abs/2201.11903

Prompt Engineering Techniques (Prompting Paradigms)



https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3

02 Transcending Input-Output Prompts

Chain-of-Thought (CoT)

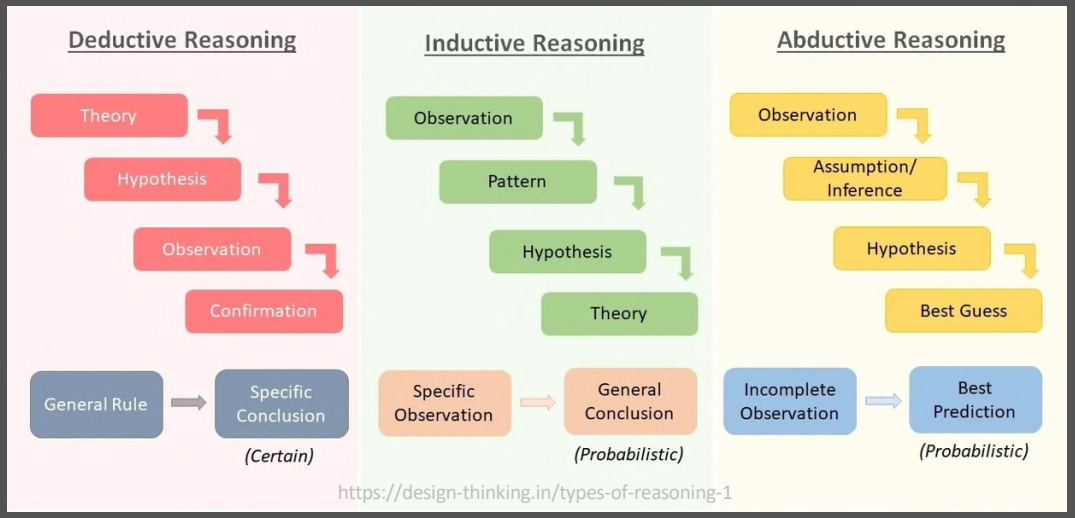
- Introduction of how a chain-of-thought works
- Variations of simple and complex chains-of-thought
- Pros and cons of chains-of-thought prompting

“A chain of thought is a series of intermediate natural language reasoning steps that lead to the final output…” (Wei et al., 2022)

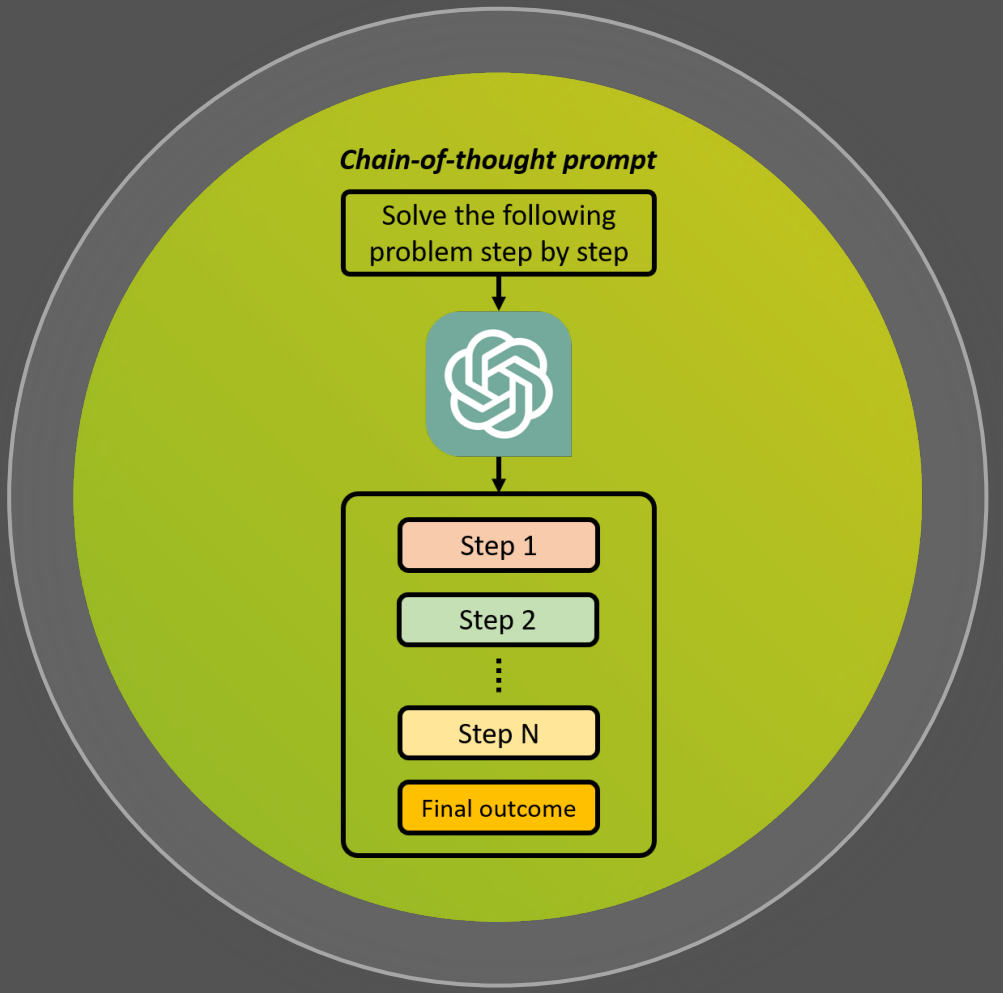
<Input, CoT, Output>

02 Transcending Input-Output Prompts

Chain-of-Thought (CoT)



https://design-thinking.in/types-of-reasoning-1



https://campus.datacamp.com/courses/chatgpt-prompt-engineering-for-developers/advanced-prompt-engineering-strategies?ex=10

Components of a CoT Rationale

- **Bridging Objects:** the symbolic items which the model traverses to reach a final conclusion. Bridging can be made up of numbers and equations for arithmetic tasks, or the names of entities in factual tasks.

Arithmetic Reasoning	Multi-hop QA
Q: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?	Q: Who is the grandchild of Dambar Shah?
A: 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.	A: Dambar Shah (? - 1645) was the father of Krishna Shah. Rudra Shah was the child of Krishna Shah (? - 1661). So the final answer (the name of the grandchild) is: Rudra Shah.

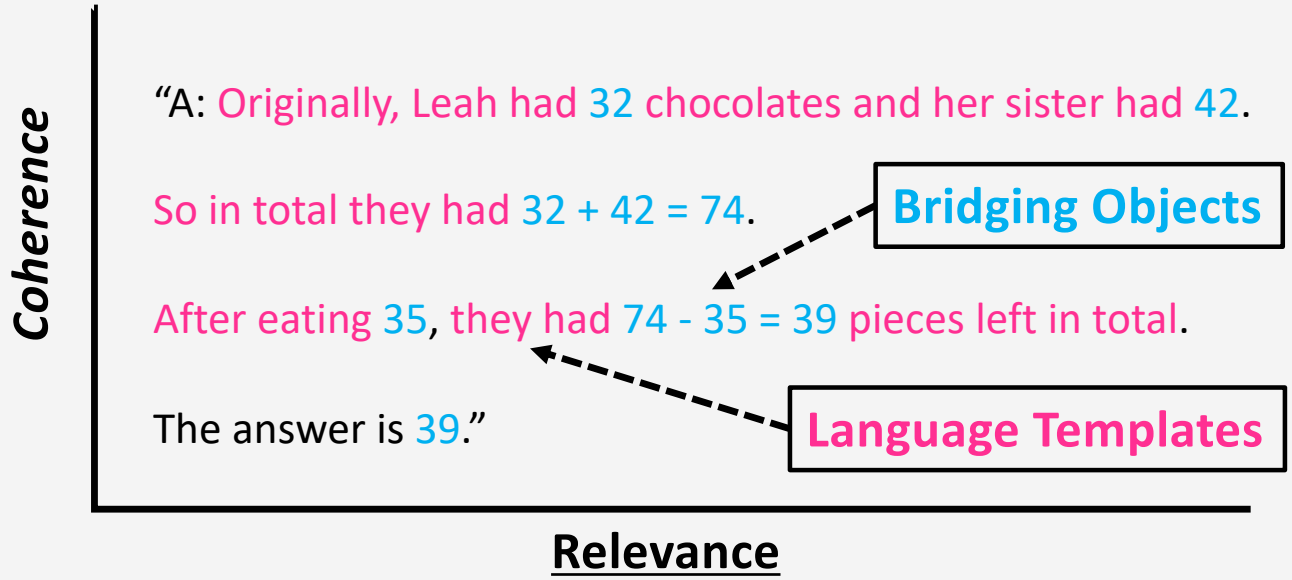
- **Language Templates:** the textual hints (relations / predicates) that guide the language model to derive and contextualize the correct bridging objects during the reasoning process.

"...being relevant [**relevance**]to the query and correctly ordering [***coherence***] the reasoning steps are the key for the effectiveness of CoT prompting" (Wang et al., 2023)

(Wang et al., 2023) https://aclanthology.org/2023.acl-long.153.pdf

"Q: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?"

"Originally, Leah had 32 + 42 = 74 chocolates and her sister had 32. So in total they had 74- 35 = 39."



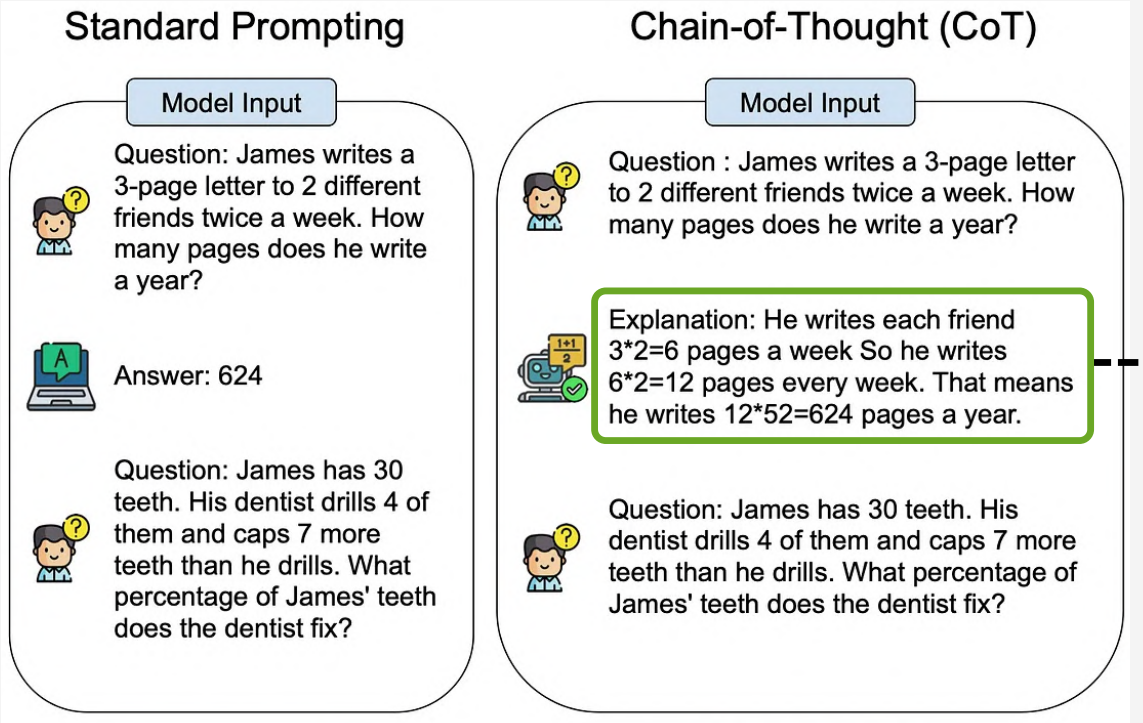
"After eating 32, they had 42 pieces left in total. Originally, Leah had 32 chocolates and her sister had 42."

"Originally, Leah had 19 chocolates and her sister had 31. So in total they had 19 + 31 = 50."

"Patricia needs to donate 32 inches and wants her hair to be 42 inches long after donating."

Relevance matters more than coherence for bridging objects and Coherence of language templates is important.

Few-Shot Chain-of-Thought Prompting



Question. What is the sentiment of the following text: 'I didn't see your internal link, I put it back. It isn't orphaned, what are you talking about?'

Answer: Negative. The answer is negative because 'what are you talking about' is passive aggressive and rude.

Question. What is the sentiment of the following text: Greetings, and thank you for your cogent remarks at. Would you be interested in voting in the straw poll as well?

Answer: Positive. The answer is positive because 'greetings' and 'thank you' are nice things to say.

Question. What is the sentiment of the following text: {text sample}

Answer:

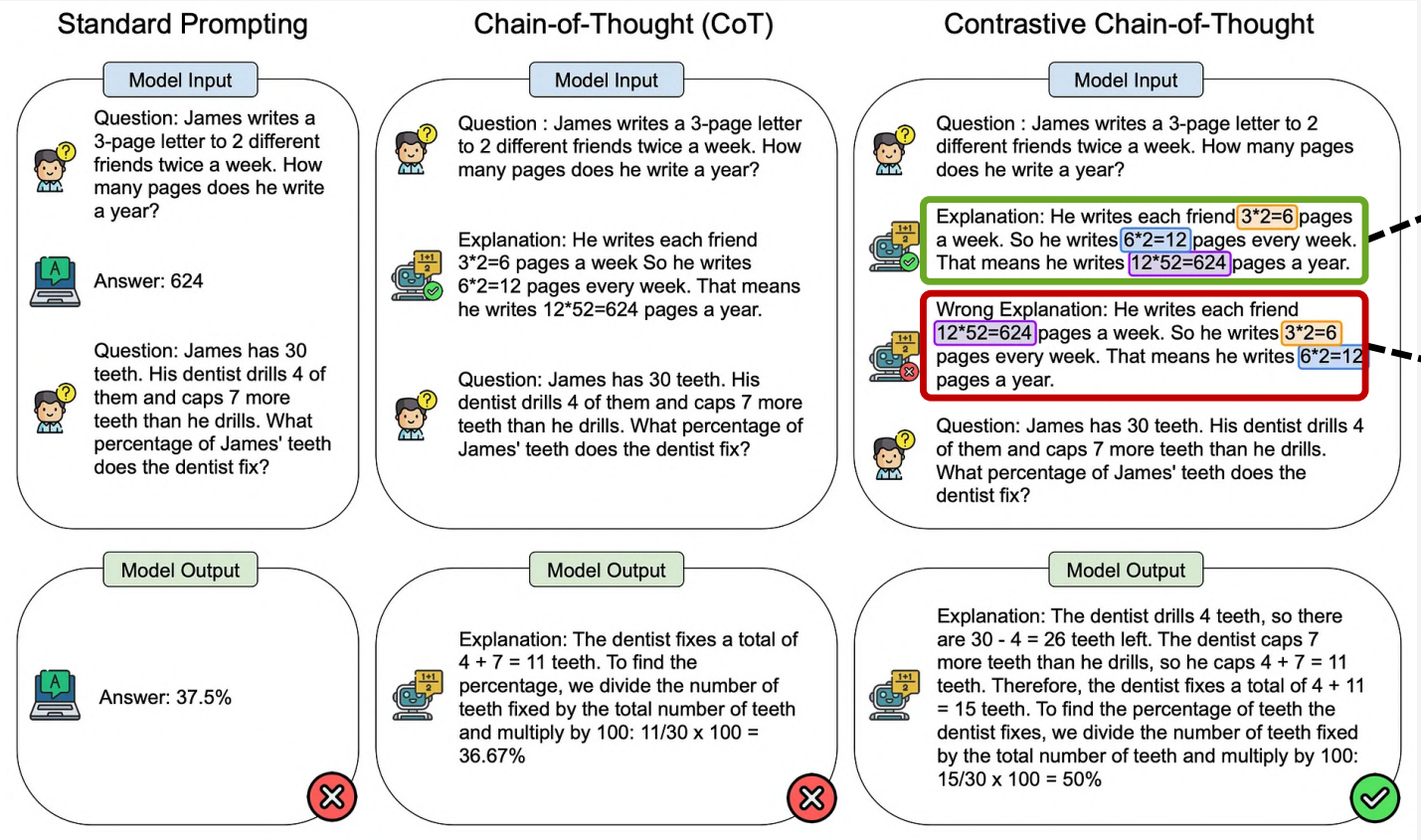
Positive CoT Example(s)

- Example 1
- Example 2
- Example 3
- ...

✓ **Increase the diversity of effective CoT examples**

✗ **Still suffer from invalid reasoning**

Contrastive Chain-of-Thought Prompting



Positive CoT Example(s)

+

Negative CoT Example(s)

✓ **Reduce vulnerability to invalid reasoning demonstrations**

✗ **Requiring human-annotated examples?**

(Chia et al., 2023) https://huggingface.co/papers/2311.09277

Zero-Shot Chain-of-Thought Prompting

"Let's think step by step."

(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)

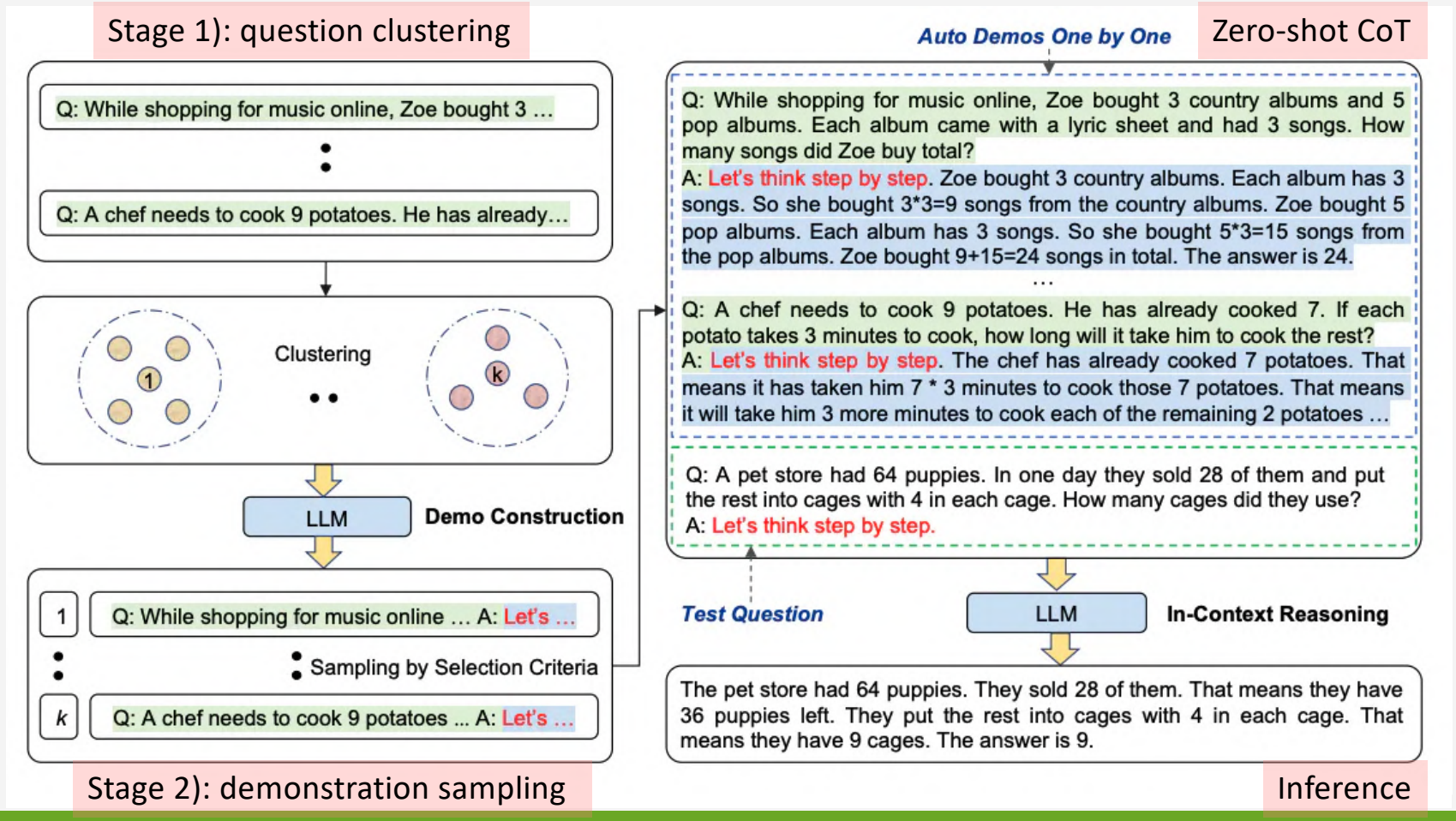
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let's think step by step.***

(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. ✓

✓ **Mitigate the trouble of coming up with (too many) examples**

✗ **May not work well in all cases**

Automatic Chain-of-Thought Prompting (Auto-CoT)



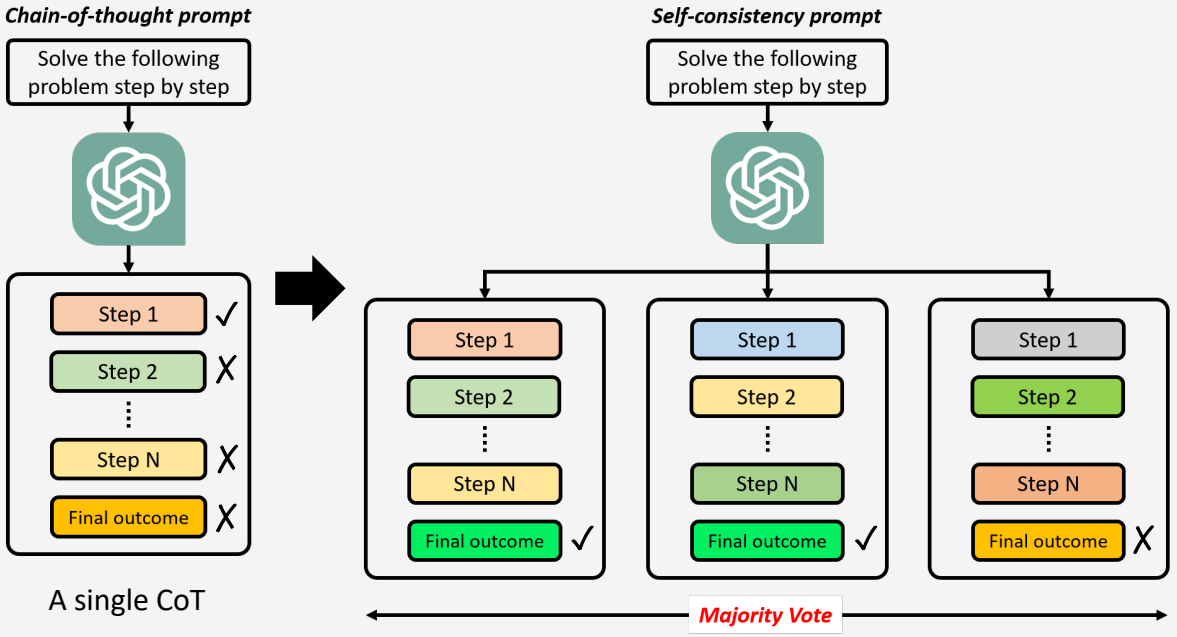
✓ **Auto-generation of positive (and negative) examples**

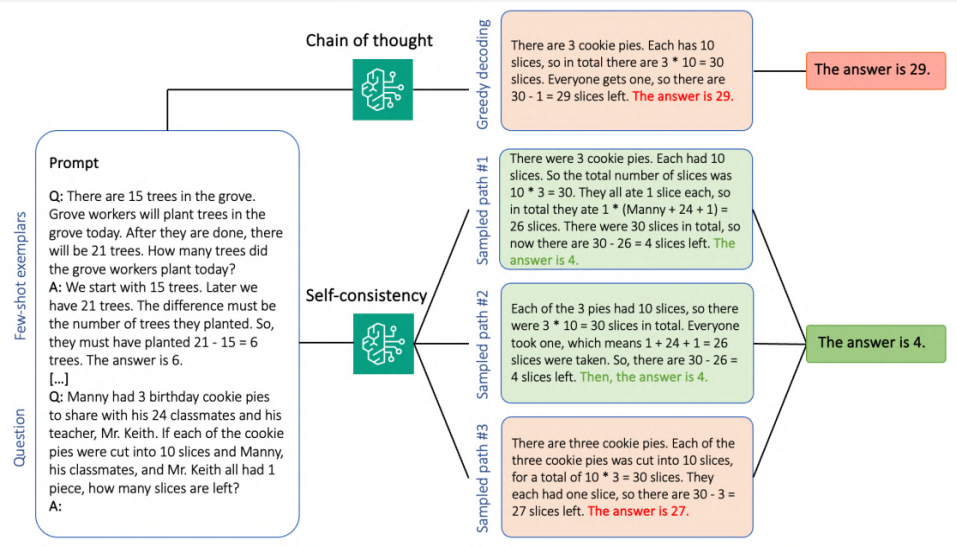
✗ **Depend on the outputs of Stage 1&2**

Self-Consistency Chain-of-Thought Prompting

Manually or through special prompt, e.g.,

"Imaging X independent experts who reason differently…"





✓ **Improve the consistency of an existing LLM without altering its architecture**

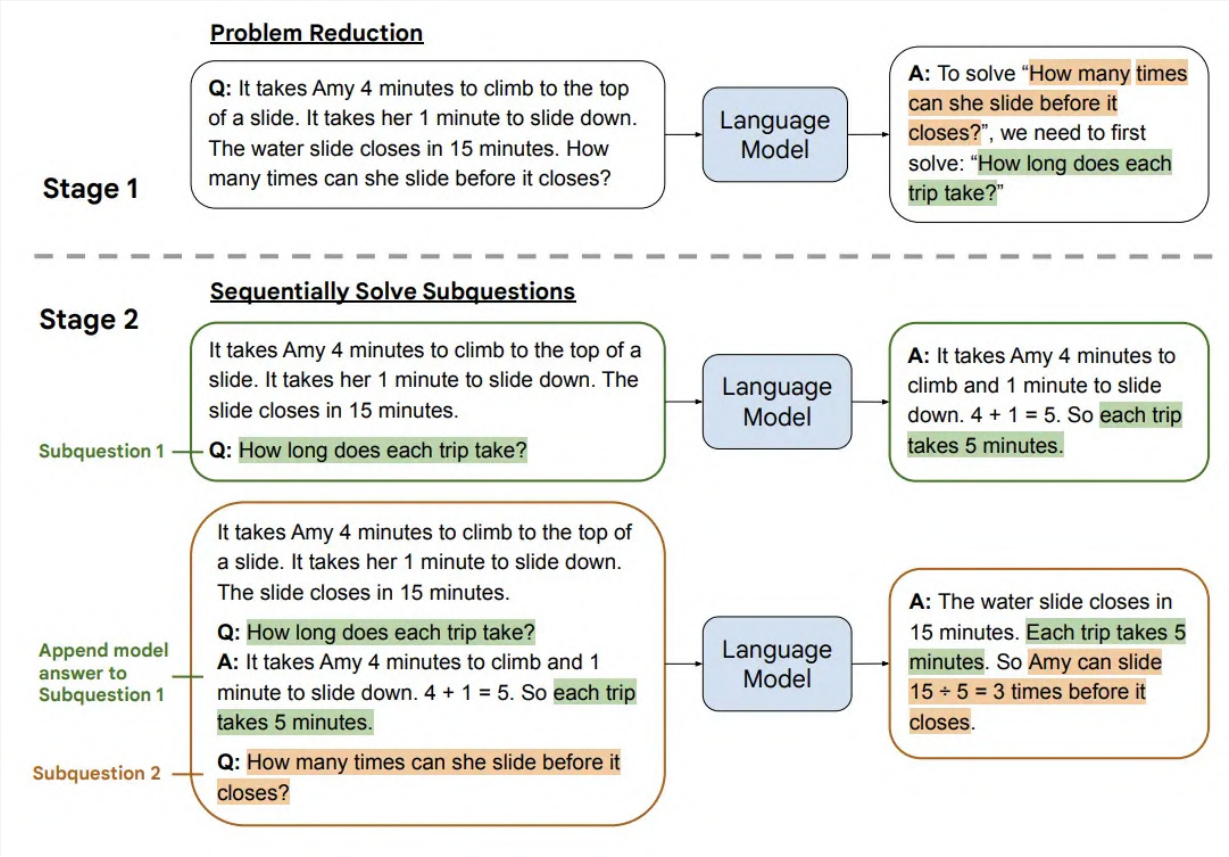
✓ **May leverage different LLMs and multi-agent setup**

✗ **Require large output tokens**

Least to Most (Chain-of-Thought) Prompting

"What subproblems must be solved before answering the inquiry?."

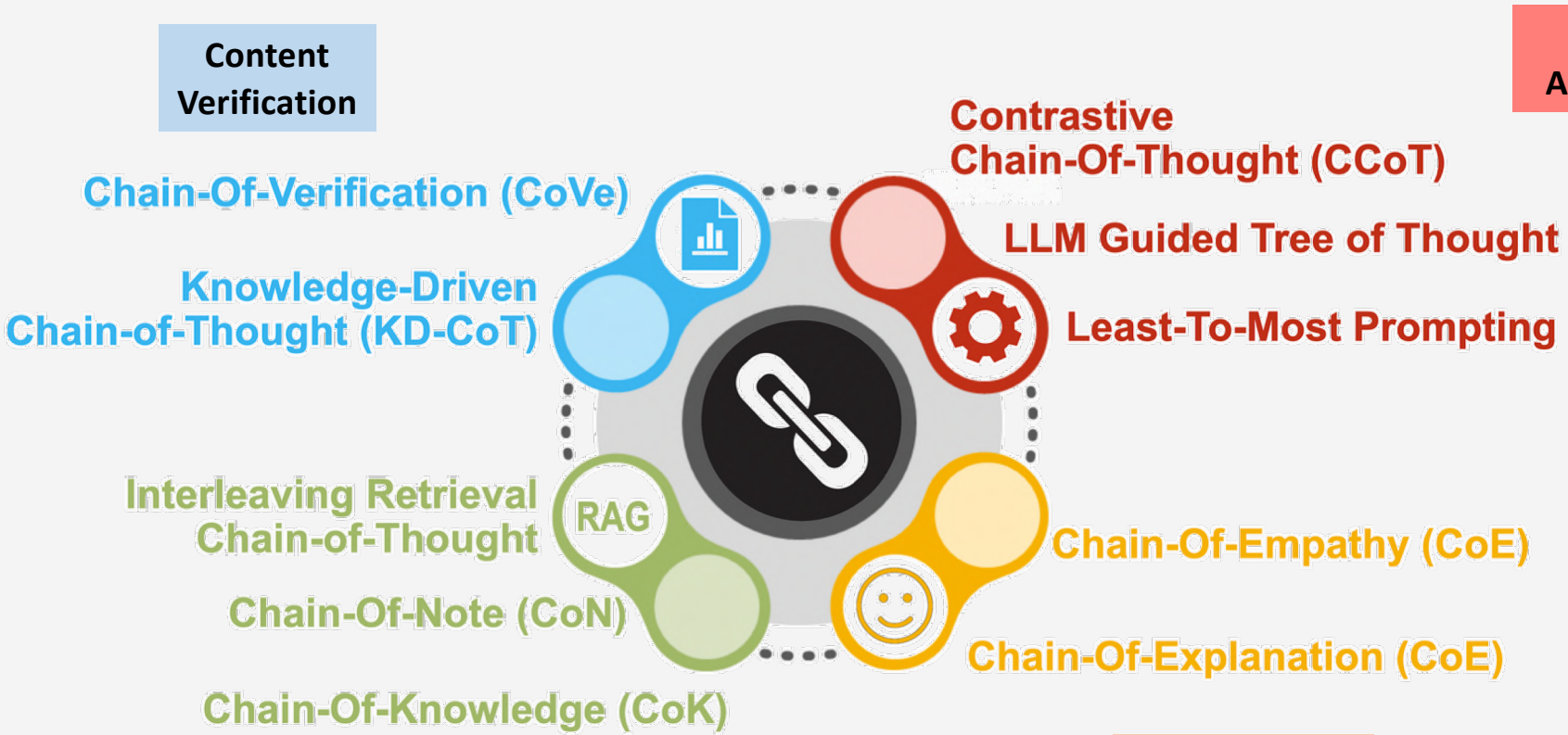
These subproblems are solved one by one. The solution of previous subproblems is fed into the prompt trying to solve the next problem.



✓ **Mitigate the trouble of coming up with (too many) examples**

✗ **May not work well in all cases**

Existing (and Ever-Growing) Family of Chain-of-X Prompting



https://cobusgreyling.medium.com/the-anatomy-of-chain-of-thought-prompting-cot-b7489c925402

CoT Vulnerabilities

- Hallucination
- Error propagation
- Lack of external knowledge
- "CoT only yields performance gains when used with models of ~100B parameters"

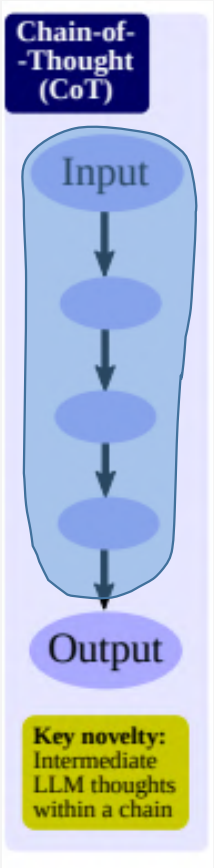
CoT Key Aspects

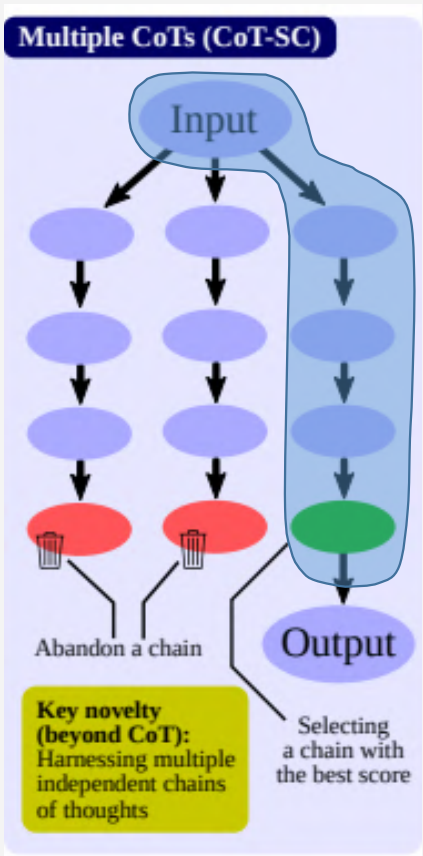
- Self-consistency
- Sensitivity to prompt design
- Coherence of the order of steps in a CoT rationale

Assuming a **fixed thought size** (#tokens) and a **fixed context size *N*** (#thoughts in the LLM context)

Latency: Number of steps between the input and output

Volume: For a thought t, the number of preceding thoughts that could impact t



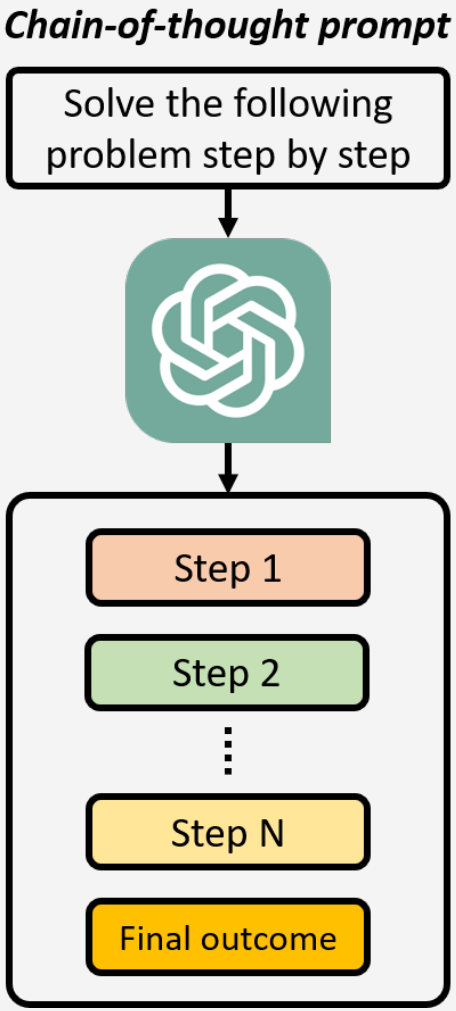


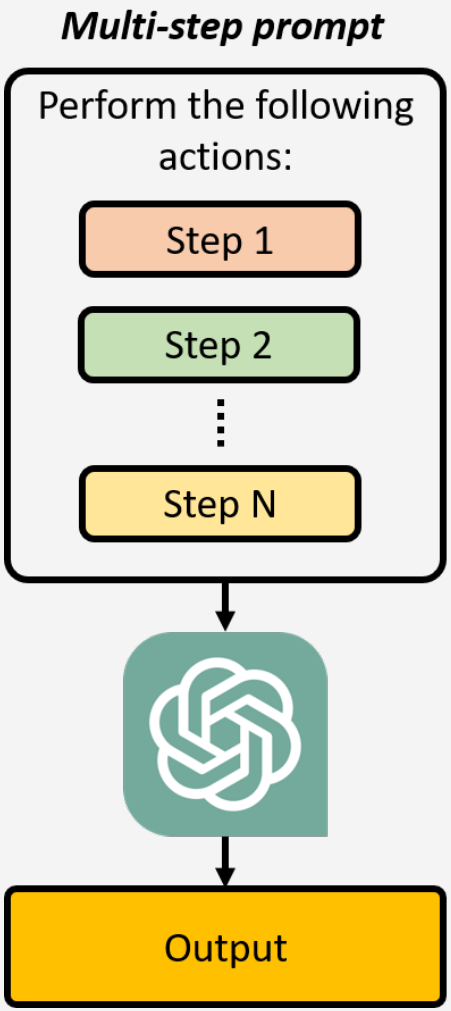
✓✓ **Large Volume: *N***	✗ **Low Volume: *N/k***
✗✗ **Large Latency: *N***	✓ **Low Latency: *N/k***

https://www.youtube.com/watch?v=n10_cdSGjXY

Image Source: (Besta et al., 2023) https://arxiv.org/pdf/2401.14295

- Sequential prompts that build upon each other
- Each subsequent prompt relies on the previous response
- Key emphasis on maintaining a coherent and continuous context and logic

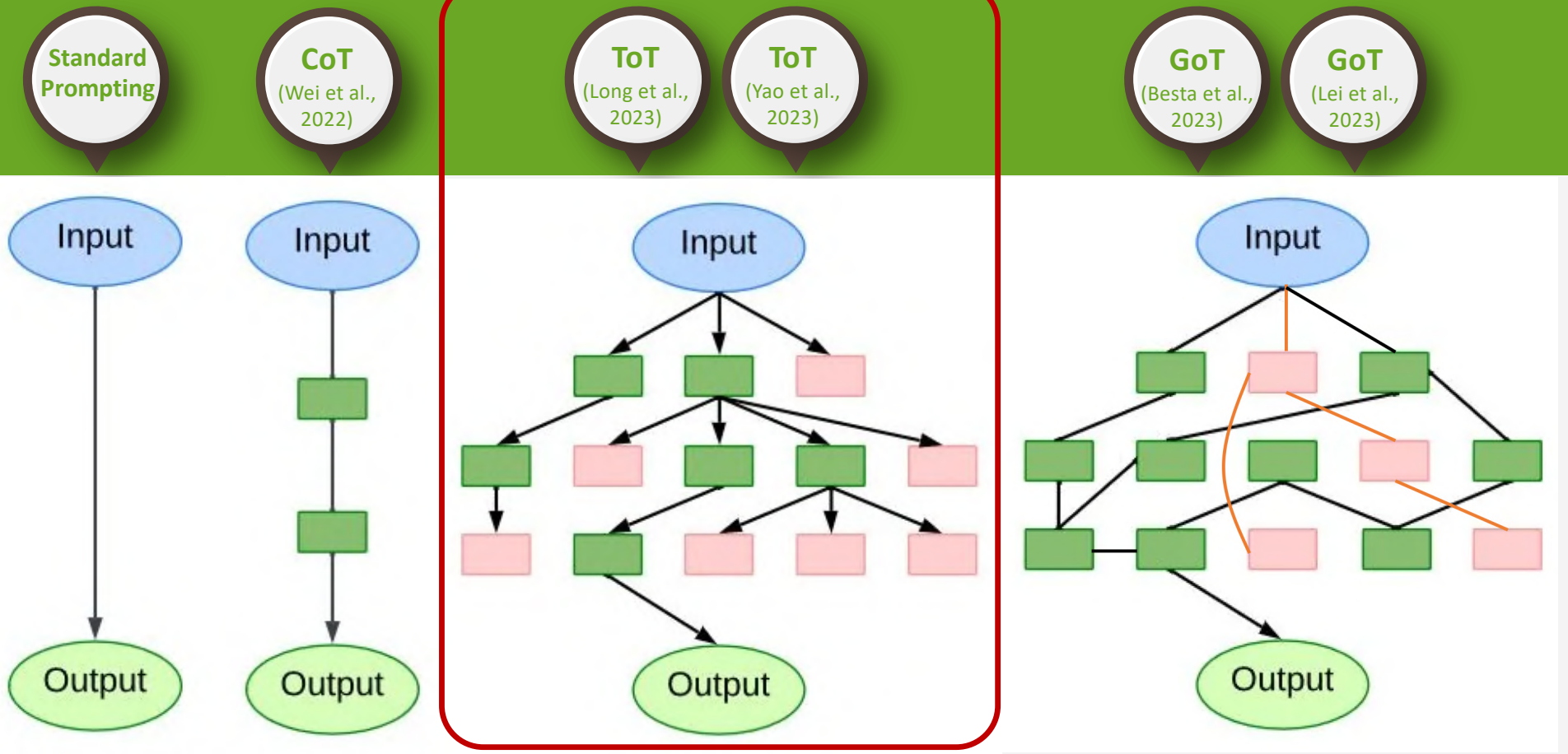




- Series of prompts designed to guide LLMs through a multi-step process of sequence
- Each step focuses on specific objectives or tasks
- Less emphasis on maintaining continuity across steps

https://campus.datacamp.com/courses/chatgpt-prompt-engineering-for-developers/advanced-prompt-engineering-strategies?ex=10

Prompt Engineering Techniques (Prompting Paradigms)



https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3

"LMs … might benefit from augmentation by a more deliberate 'System 2' planning process that … explores diverse alternatives … and evaluates its current state" (Yao et al., 2023)

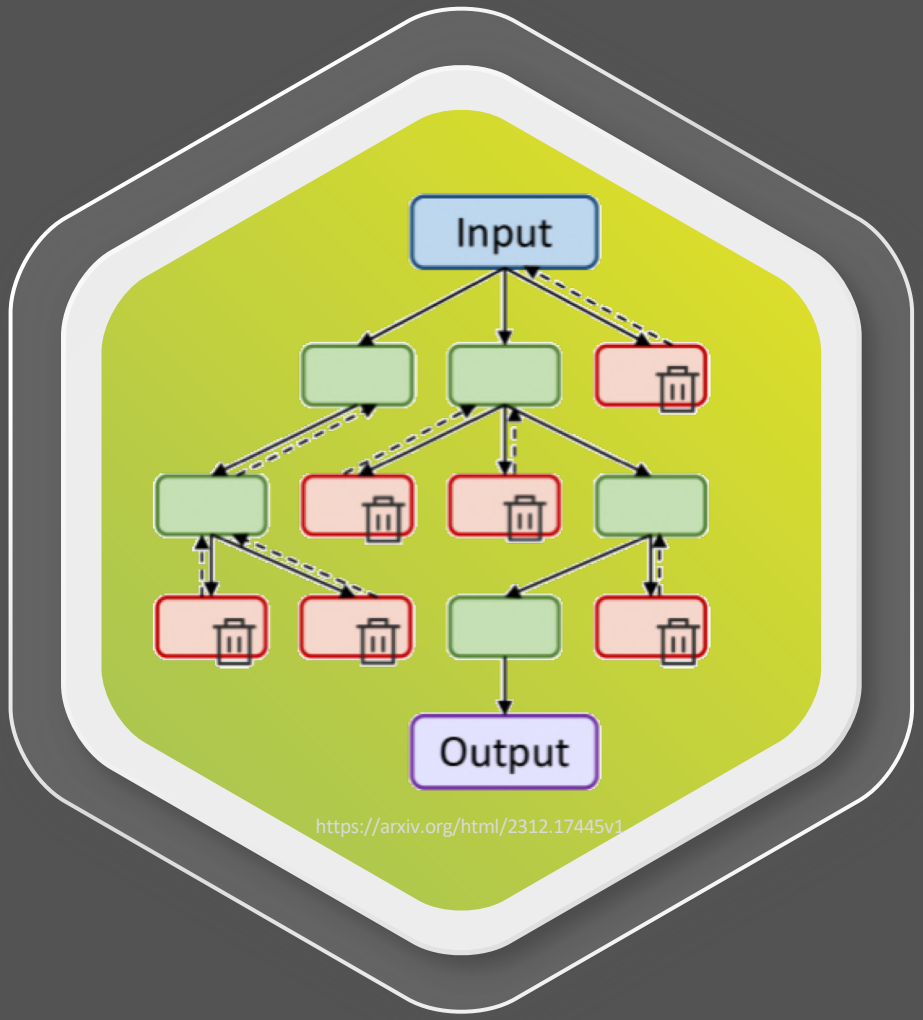
03

Deliberate Problem Solving

Tree-of-Thoughts

- Description of how a tree-of-thoughts works
- Concepts and mechanisms of trees-of-thoughts
- Pros and cons of using trees-of-thoughts

System 1	System 2
• Fast • Automatic • Non-conscious • Common Use • 'Good Enough'	• Slow • Deliberative • Conscious • Requires Effort • Use If Critical



03

Deliberate Problem Solving

Tree-of-Thoughts

Decomposition — Breaks a problem into a sequence of smaller steps—or thoughts—that are solved individually

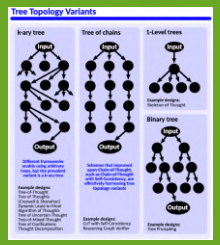
Generation — Explore multiple choices for each problem-solving thought.

Evaluation — Evaluate whether certain thoughts bring the model closer to a final solution

Search — Search over a combinatorial space of possible problem-solving steps to find the best final solution. Perform backtracking when certain thoughts are found to be a dead end.

1) Thought Decomposition: Break Intermediate Process into Steps

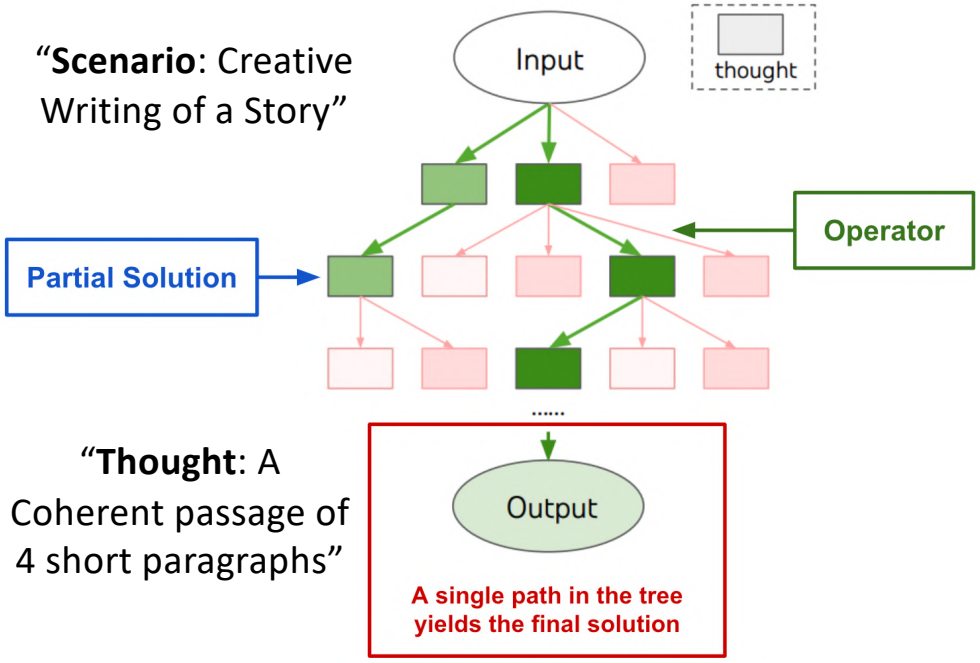
- Definition of "thought" is task dependent and may take different forms



"A thought … should be **small enough** so that LMs can generate **promising and diverse samples** …"

E.g., generating a whole book is usually too big to be coherent

"**Scenario**: Creative Writing of a Story"



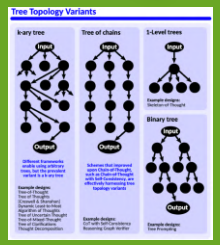
"**Thought**: A Coherent passage of 4 short paragraphs"

"A thought … should be … yet **big enough** so that LMs can **evaluate its prospect** toward problem solving …"

E.g., generating one word in a paragraph is usually too small to evaluate

1) Thought Decomposition: Break Intermediate Process into Steps

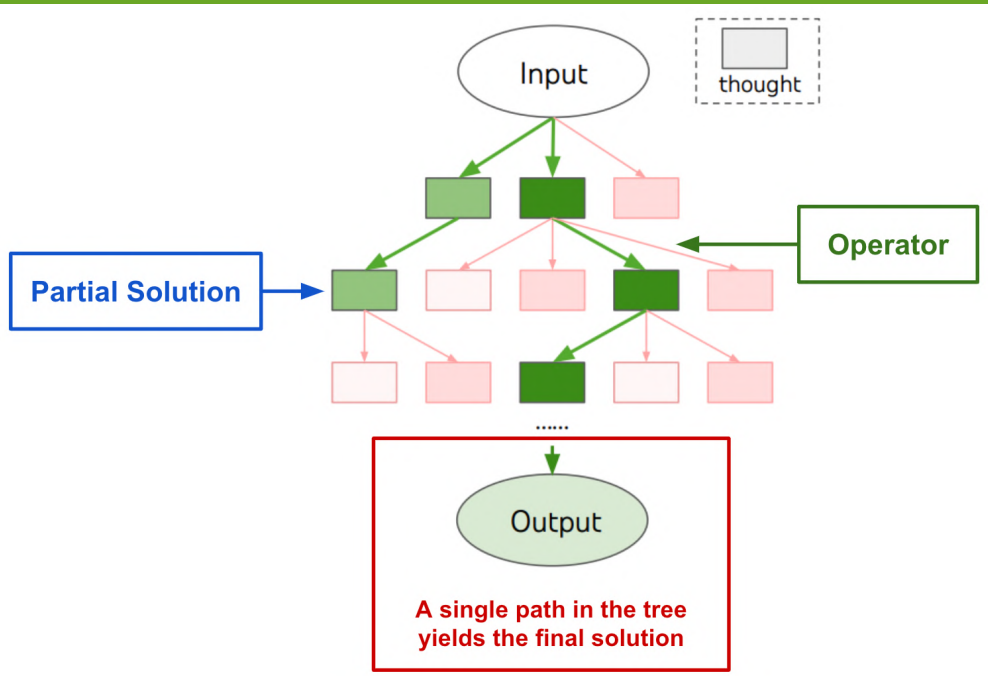
- Definition of “thought” is task dependent and may take different forms
- The “thoughts” in the tree can be homogeneous or heterogeneous
- The topology of the tree (depth, degrees, etc.) may vary based on the decomposition



“**Scenario**: Sudoku”

- ➢ Insert a number
- ➢ Insert a number
- ➢ Insert a number

Each level has the same purpose; the number of sub-branches is determined by the rule.



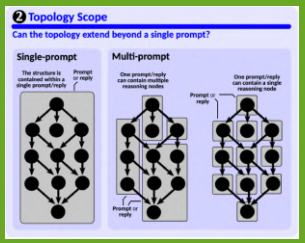
“**Scenario**: Design a marketing strategy for a new product launch”

- ➢ Identify target audience
- ➢ Evaluate competitor and trend
- ➢ Design promotional campaign

Each level has a different purpose; the number of sub-branches under each node is not fixed and can be pre-defined.

2) Thought Generation: Generate K Candidates Given a Tree State

- Definition of “thought” is task dependent and may take different forms
- **Sampling**: generating several thoughts independently with the same prompt
- **Proposing**: generating several thoughts sequentially with a “propose prompt”



The sampling approach works best when the thought space is rich, as several independently-generated thoughts are unlikely to be duplicates.

Thought Sampling

Perform the following actions:

Step 1

Step 2

⋮

Step N

“**Scenario**: Creative Writing”

E.g., each thought is a paragraph

$$z^{(j)} \sim p_\theta^{CoT}(z_{i+1}|s) = p_\theta^{CoT}(z_{i+1}|x, z_{1\cdots i}) \quad (j = 1 \cdots k)$$

Output 1 | Output 2 | ... | Output K

“**Scenario**: Game of 24”

E.g., each thought is just a word or a line

Thought Proposing

Perform the following actions:

Step 1

Step 2

⋮

Step N

$$[z^{(1)}, \cdots, z^{(k)}] \sim p_\theta^{propose}(z_{i+1}^{(1\cdots k)} \mid s)$$

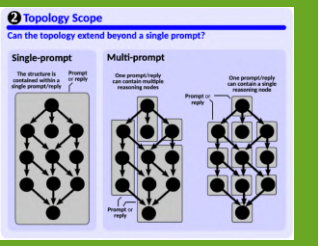
Output 1 | Output 2 | ... | Output K

If the thought space is more constrained, then the proposing technique can be used to generate several thoughts in the same context while avoiding duplicates.

(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://www.semanticscholar.org/reader/eff9d7ed06f30f121d30ee13802a11f172ef66f4
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting

2) Thought Generation: Generate K Candidates Given a Tree State

- Definition of “thought” is task dependent and may take different forms
- **Sampling**: generating several thoughts independently with the same CoT prompt
- **Proposing**: generating several thoughts sequentially with a “propose prompt”



Sampling or Proposing?

Thought Sampling

Prompt: *Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realizes they're wrong at any point, then they leave. The question is…*

Output 1 | Output 2 | … | Output K

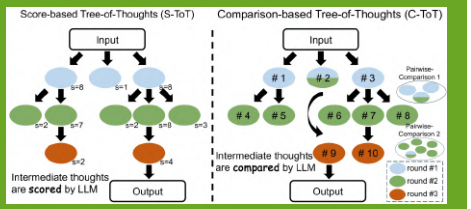
Thought Proposing

Output 1 | Output 2 | … | Output K

(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://www.semanticscholar.org/reader/eff9d7ed06f30f121d30ee13802a11f172ef66f4
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting
https://medium.com/@kyeg/unleashing-super-intelligence-with-tree-of-thoughts-f2f744786e65

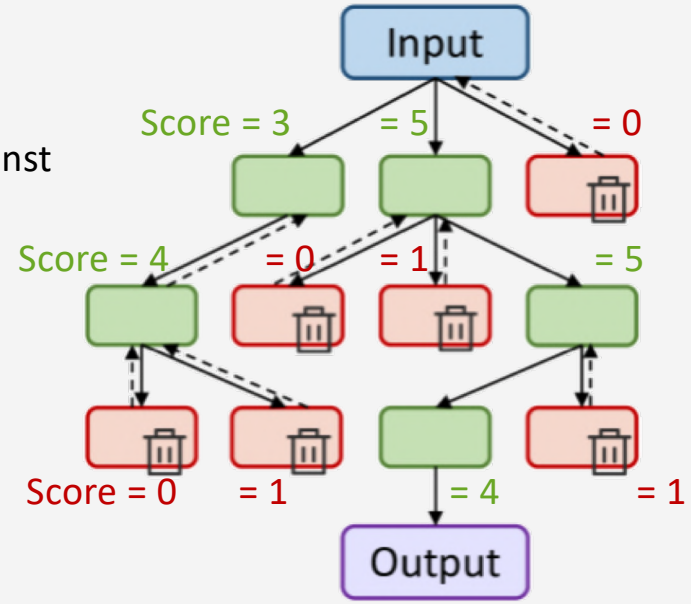
3) State Evaluation: Deliberate Reasoning of Progress to a Solution

- Define a “heuristic” for evaluation: a set of criteria or a holistic measure
- **Value:** independently assign a scalar value or classification to each state



Score-based:

- Value formula
- Violations against given criteria
- Distance to success

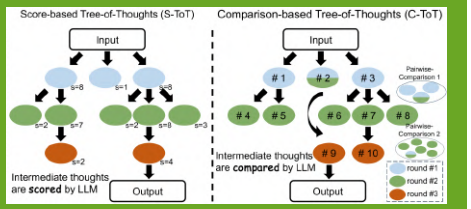



```
1. Write a story featuring Mowgli.
    *Choice 1: Mowgli is lonely and wants friends.
    *Choice 2: Mowgli is adventurous and
      wants to explore the villages.
    *Choice 3: Mowgli is curious and wants
      to learn about the human kingdom
2. Pick one choice and keep writing the story.
3. Check your story for how creative, clear,
   and well-written it is.
```

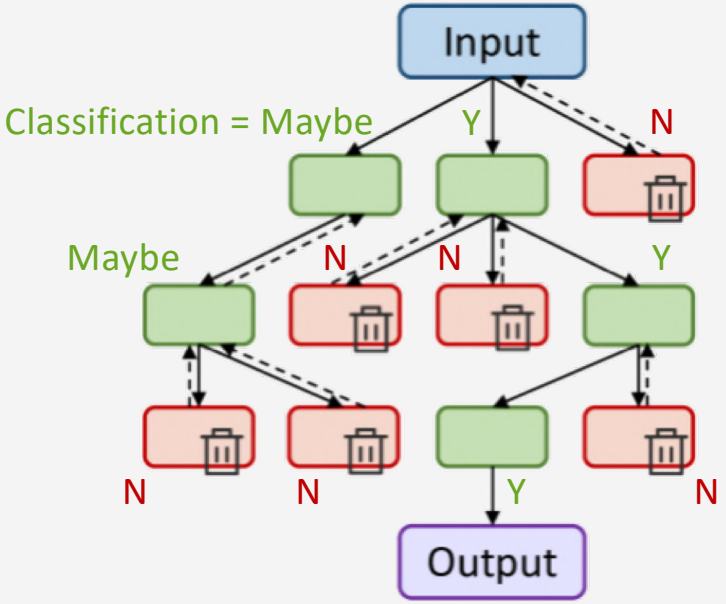
(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://arxiv.org/html/2312.17445v2
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting
https://dev.to/zokizuan/tree-of-thoughts-a-new-way-to-prompt-ai-2dle

3) State Evaluation: Deliberate Reasoning of Progress to a Solution

- Define a "heuristic" for evaluation: a set of criteria or a holistic measure
- **Value:** independently assign a scalar value or classification to each state
  - Current state only vs. look-ahead simulation
  - Promoting "good" states and/or eliminating "bad" states




```
1. Write a story featuring Mowgli.
    *Choice 1: Mowgli is lonely and wants friends.
    *Choice 2: Mowgli is adventurous and
      wants to explore the villages.
    *Choice 3: Mowgli is curious and wants
      to learn about the human kingdom
2. Pick one choice and keep writing the story.
3. Check your story for how creative, clear,
   and well-written it is.
```

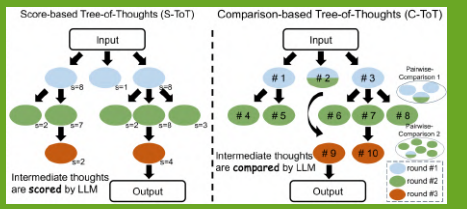


Classification-based:

- Categorical classifier
- Ordinal scale that could be heuristically converted into values

3) State Evaluation: Deliberate Reasoning of Progress to a Solution

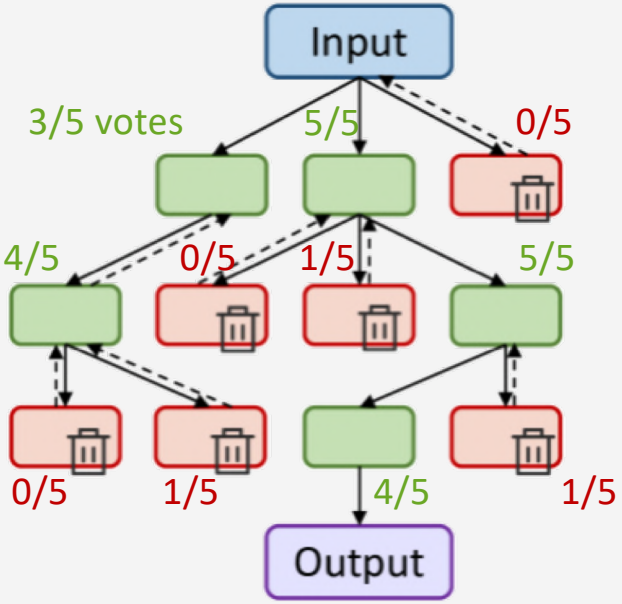
- Define a "heuristic" for evaluation: a set of criteria or a holistic measure
- **Value:** independently assign a scalar value or classification to each state
- **Vote**: compare different solutions and select the one that is most promising



Comparison-based:

- Pair-wise comparison
- Ranking
- Top K
- …

An example of pair-wise tournament: "Compare thoughts a and b by LLM n times based on the given needs, then take a majority vote. If a wins, keep thought a in the pool and drop b, and vice versa."

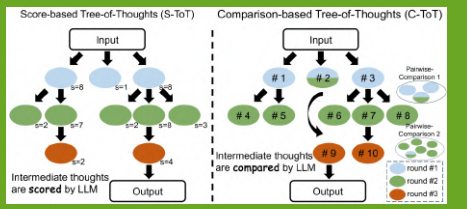


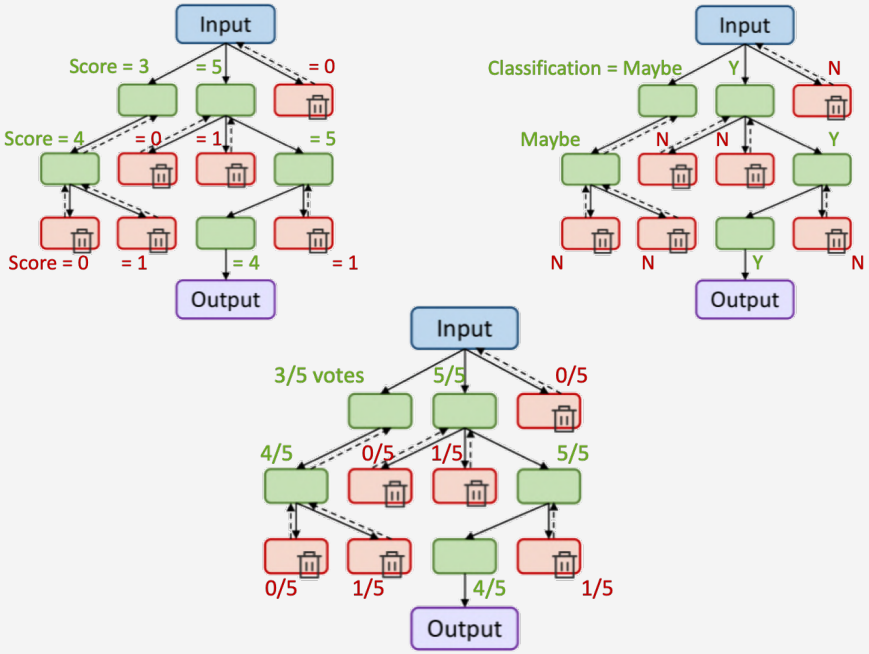

```
1. Imagine you're trying to find a good place to go on vacation.
   Your needs are:
   *Find a good vacation spot. Keep in mind:
   *Must be in India.
   *Good for families.
   *Has fun outdoor activities.
   *Has interesting historical places.
   *Summer weather should be bearable.

2. Choices for looking up places:
   *Choice 1: Look at places in Northern India.
     *More choices: Himachal Pradesh or Uttarakhand.
   *Choice 2: Look at places in Southern India.
     *More choices: Kerala or Tamil Nadu.
   *Choice 3: Look at places in Western India.
     *More choices: Goa or Rajasthan.

3. Pick a choice and dig deeper.
4. Compare all the places you found based on your needs.

Here, you can use a 'browser' tool to help you look for
info along the path you chose. The AI can follow a path,  do
the searching, and check how good the options are based on
what you need.  This way of solving problems can give you more
complete and fitting answers.
```

(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://arxiv.org/html/2312.17445v2
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting
https://dev.to/zokizuan/tree-of-thoughts-a-new-way-to-prompt-ai-2dle

3) State Evaluation: Deliberate Reasoning of Progress to a Solution

- Define a “heuristic” for evaluation: a set of criteria or a holistic measure
- **Value:** independently assign a scalar value or classification to each state
- **Vote**: compare different solutions and select the one that is most promising





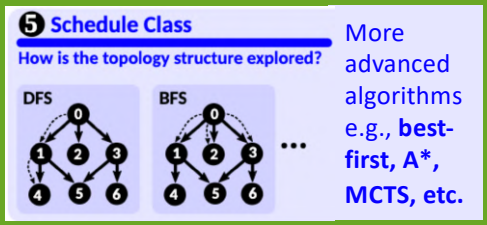
Voting is best when a successful solution to a problem is hard to directly value (e.g., creative writing tasks), e.g., casting “which state to explore” as a multi-choice QA by comparing different partial solutions

For both methods, could prompt the LM multiple times to aggregate the value or vote results to trade time/resource/cost for more faithful/robust heuristics

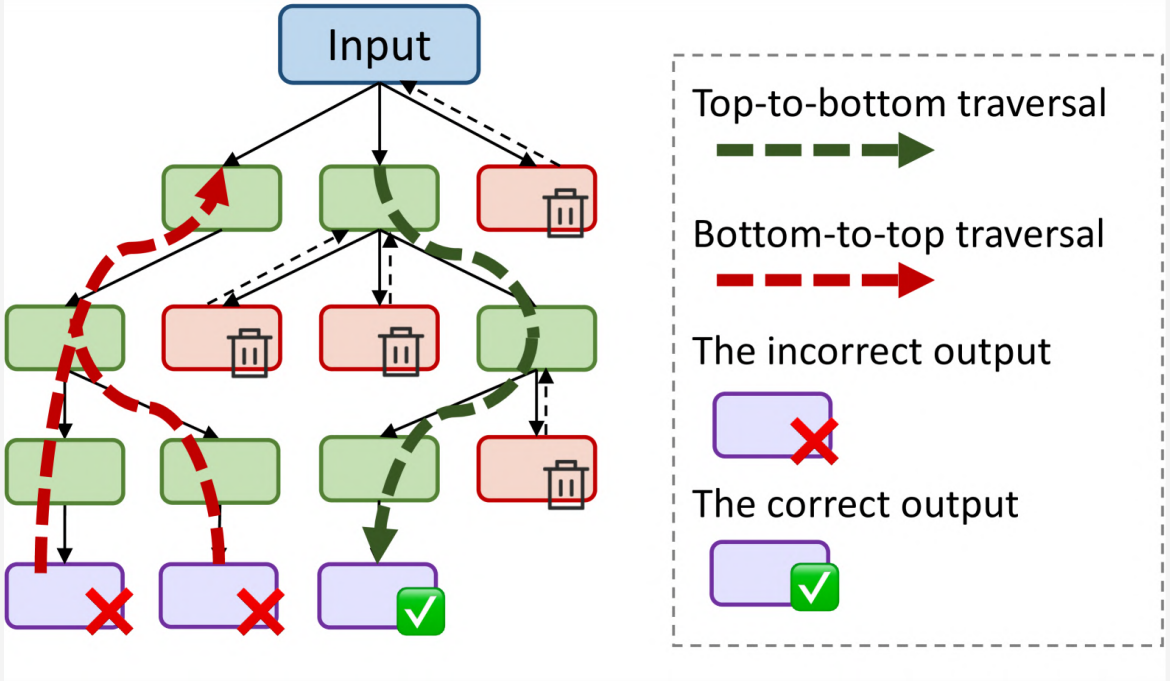
(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://arxiv.org/html/2312.17445v2
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting

4) Search Algorithm: Explore the Solution Space

- Plug and play a proper tree search algorithm based on the tree structure



Back-tracking to a preceding state when the outcome is not satisfactory or hit the limit

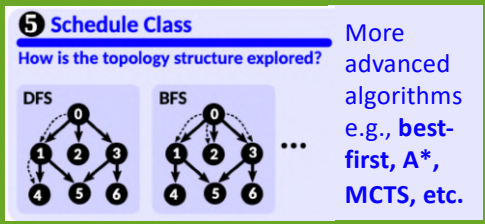


Exploring forward to identify a feasible, hopefully optimal path to the final solution

(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://arxiv.org/html/2312.17445v2
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting
https://www.semanticscholar.org/reader/eff9d7ed06f30f121d30ee13802a11f172ef66f4

4) Search Algorithm: Explore the Solution Space

- Plug and play a proper tree search algorithm based on the tree structure
- **BFS:** maintaining a set of the *b* most promising states per step
  - Local vs. global evaluation at each layer
  - Set b to a small number to prune the tree



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 $t = 1, \cdots, T$ **do**

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

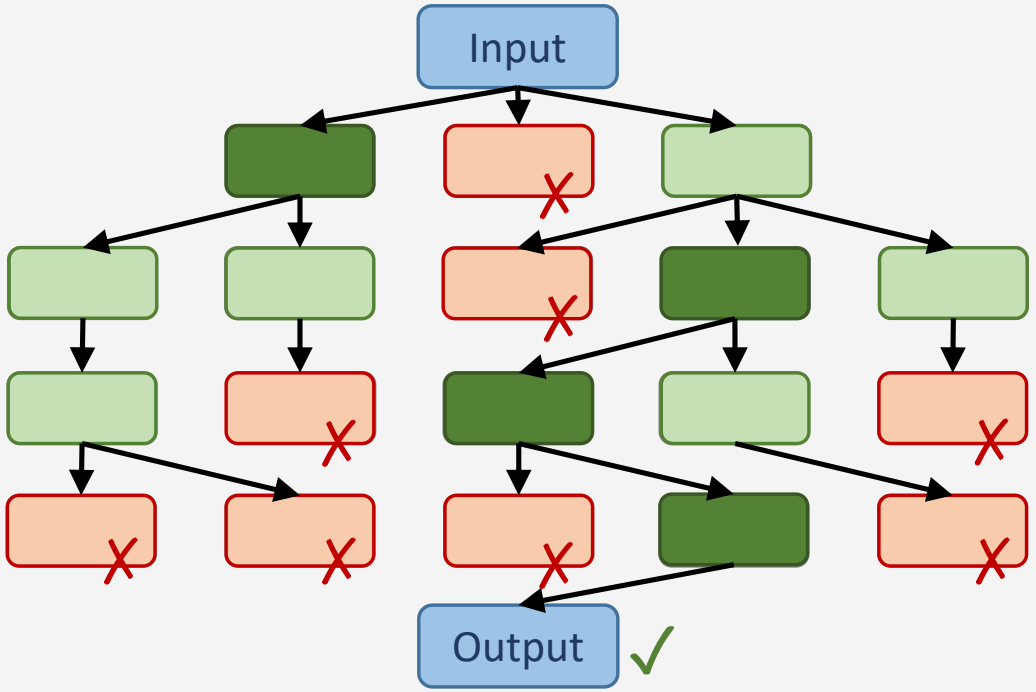
$V_t \leftarrow V(p_\theta, S'_t)$

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

end for

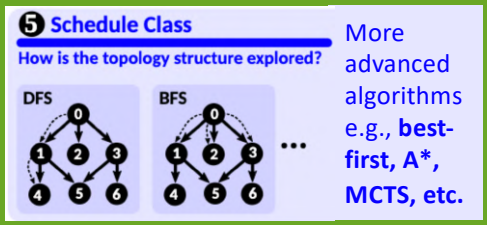
return $G(p_\theta, \arg\max_{s \in S_T} V_T(s), 1)$

It is an excellent choice when the depth of the tree is relatively small, and solutions are spread out evenly



4) Search Algorithm: Explore the Solution Space

- Plug and play a proper tree search algorithm based on the tree structure
- **BFS:** maintaining a set of the *b* most promising states per step
- **DFS**: exploring the most promising state first
  - Until the final output is reached or state evaluator find the thought impossible to succeed

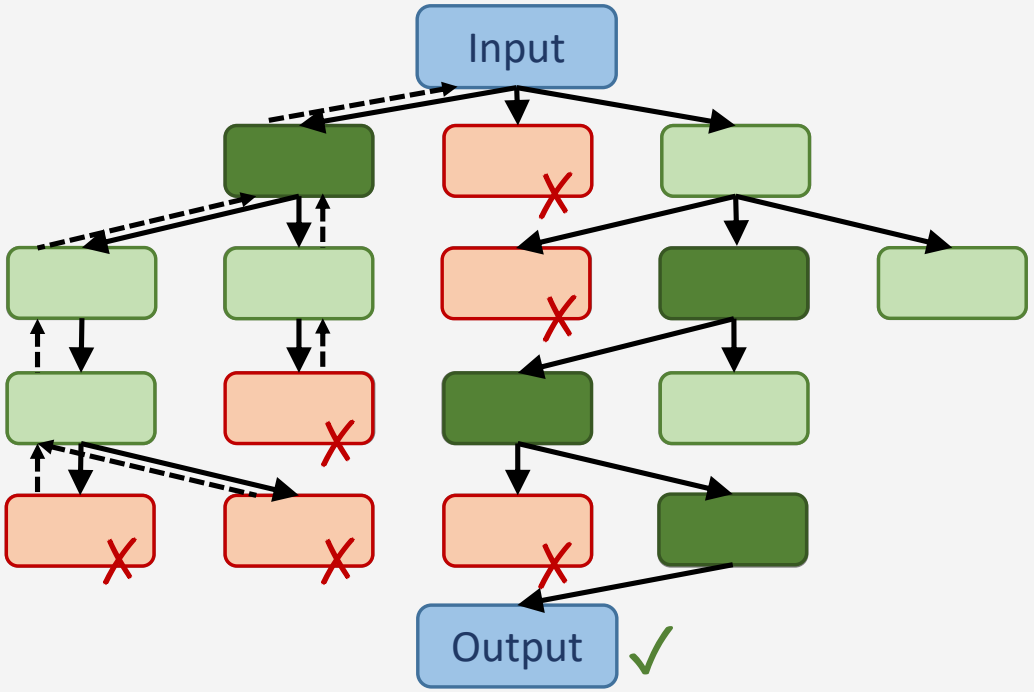


Algorithm 2 ToT-DFS$(s, t, p_\theta, G, k, V, T, v_{th})$

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_θ, s, 1)
end if
for s' ∈ G(p_θ, s, k) do        ▷ sorted candidates
    if V(p_θ, {s'})(s) > v_thres then ▷ pruning
        DFS(s', t + 1)
    end if
end for
```

It is suitable when the tree depth is significant, and solutions are located deep in the tree



(Yao et al. 2023) https://arxiv.org/abs/2305.10601
https://arxiv.org/html/2312.17445v2
https://cameronrwolfe.substack.com/p/tree-of-thoughts-prompting
https://www.semanticscholar.org/reader/eff9d7ed06f30f121d30ee13802a11f172ef66f4

Tree-of-Thought Piecing Together

```
# Initialize an agent from swarms
agent = Agent(
    agent_name="tree_of_thoughts",
    agent_description="This agent uses the tree_of_thoughts library to generate thoughts.",
    system_prompt=None,
    llm = OpenAIChat(),
)

# Initialize the ToTAgent class with the API key
model = ToTAgent(
    agent,
    strategy="cot",
    evaluation_strategy="value",
    enable_react=True,
    k=3,
)
```

Generator & Evaluator

```
# Initialize the MonteCarloSearch class with the model
tree_of_thoughts = MonteCarloSearch(model)
```

Search Algorithm

```
# Define the initial prompt
initial_prompt = """
Input: 2 8 8 14
Possible next steps:
2 + 8 = 10 (left: 8 10 14)
8 / 2 = 4 (left: 4 8 14)
14 + 2 = 16 (left: 8 8 16)
2 * 8 = 16 (left: 8 14 16)
8 - 2 = 6 (left: 6 8 14)
14 - 8 = 6 (left: 2 6 8)
14 /  2 = 7 (left: 7 8 8)
14 - 2 = 12 (left: 8 8 12)
Input: use 4 numbers and basic arithmetic operations (+-*/) to obtain 24 in 1 equation
Possible next steps:
"""
```

Few-shot CoT Propose Prompt

```
# Define the number of thoughts to generate
num_thoughts = 1
max_steps = 3
max_states = 4
pruning_threshold = 0.5
```

Tree Topology

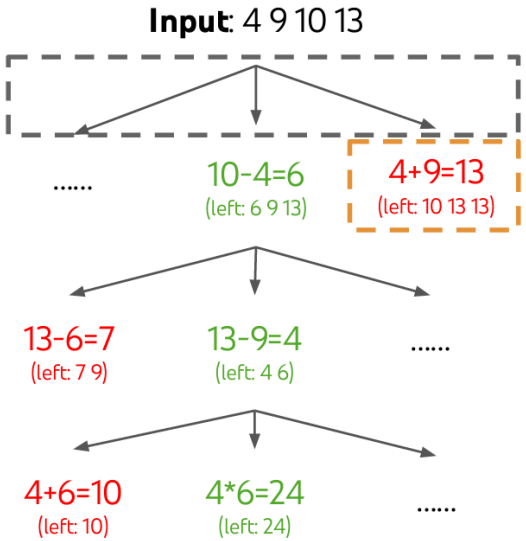
+

```
# Generate the thoughts
solution = tree_of_thoughts.solve(
    initial_prompt=initial_prompt,
    num_thoughts=num_thoughts,
    max_steps=max_steps,
    max_states=max_states,
    pruning_threshold=pruning_threshold,
    # sleep_time=sleep_time
)

print(f"Solution: {solution}")
```

https://github.com/kyegomez/tree-of-thoughts?source=post_page-----f2f744786e65----------

Tree-of-Thought Prompting Example I: Game of 24

Tree Depth: 3 (fixed)



Thought: intermediate equations and the corresponding outcome

Thought Generation: Proposing

(a) Propose Prompt

{one example}
Input: 4 9 10 13
Possible next steps:

LM

Thought Generation

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

Game of 24

(b) Value Prompt

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

LM

Thought Evaluation

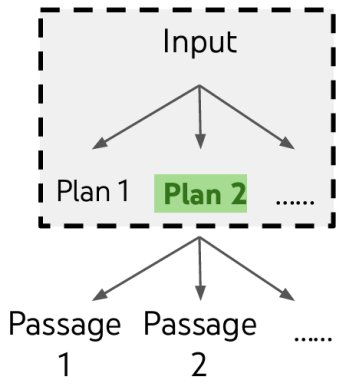
(13 - 10) * 13 = 3 * 13 = 39
10 + 13 + 13 = 36 There is no way to obtain 24 with these big numbers. impossible

State Evaluation: Value (Classification)

Search Algorithm: BFS

Tree-of-Thought Prompting Example II: Creative Writing

Tree Depth: 1 (fixed, if plan only); 2~ (varied, if including content)



Thought Generation: Sampling

(a) **Input**

Write a coherent passage of 4 short paragraphs. The end sentence of each paragraph must be: **1.** It isn't difficult to do a handstand if you just stand on your hands. **2.** It caught him off guard that space smelled of seared steak. **3.** When she didn't like a guy who was trying to pick her up, she started using sign language. **4.** Each person who knows you has a different perception of who you are.

(b) **Plans**

Plan 1

1. Introduce and explain the technique of doing a handstand **2.** Switch to a story about an astronaut's first time in space **3.** Describe a situation where a woman uses sign language to avoid unwanted attention **4.** The final paragraph explains how everyone has different perceptions of others

0/5 votes

Plan 2

1. Introduction to an unusual self-help book, mentioning a handstand as a metaphor for embracing challenges. **2.** Discuss the unexpected things learned from astronauts, including the smell of space. **3.** Describe a woman's clever tactic for avoiding unwanted attention at a bar. **4.** Contemplate how different perceptions of oneself can shape one's identity.

3/5 votes

Plan 3-5

1....
2....
...

n/5 votes

(c) **Votes**

Analyzing each choice in detail: Choice 1, while incorporating the required end sentences, seems to lack a clear connection between the paragraphs {...} Choice 2 offers an interesting perspective by using the required end sentences to present a self-help book's content. It connects the paragraphs with the theme of self-improvement and embracing challenges, making for a coherent passage. {...} **The best choice is 2.**

Thought: writing plan (and passages if including content)

State Evaluation: Vote

Search Algorithm: N/A (fixed, if plan only); DFS (varied, if including content)

Support to Problem Solving

- **Contextual depth**: providing clear direction
- **Structured exploration**: facilitating systematic analysis
- **Enhanced relevance**: tailoring responses to angles
- **Iterative refinement**: optimizing for accuracy and coherence

Other Possible Variations

- **Typology derivation**: manually defined vs. constructed by LLM
- **Typology representation**: implicit vs. explicit
- **Search representation**: textual description, algorithmic code, in-context example(s), etc.

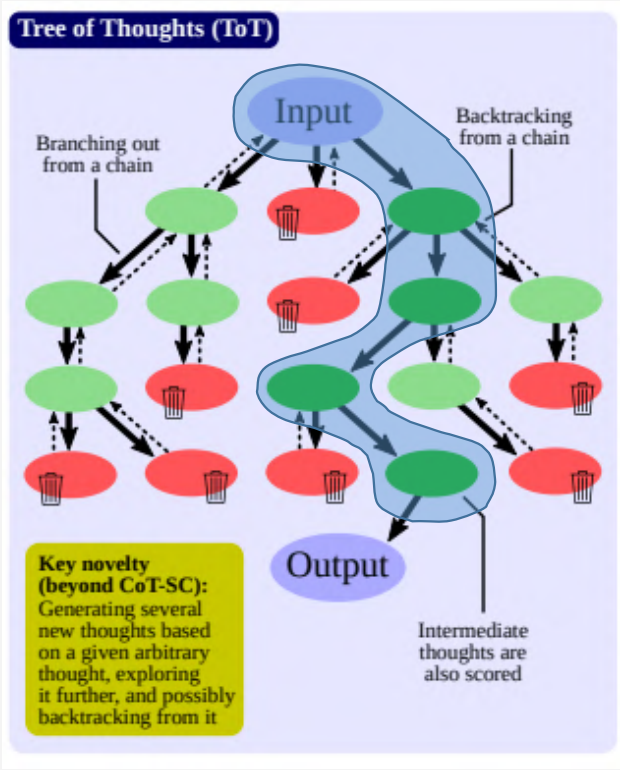
- o **Cost**
- o **Complexity**
- o **Information Load**
- o **Linear Relationship**

https://www.youtube.com/watch?v=n10_cdSGjXY
https://www.semanticscholar.org/reader/eff9d7ed06f30f121d30ee13802a11f172ef66f4
(Besta et al., 2023) https://arxiv.org/pdf/2401.14295

Assuming a **fixed thought size** (#tokens) and a **fixed context size *N*** (#thoughts in the LLM context)

Latency: Number of steps between the input and output

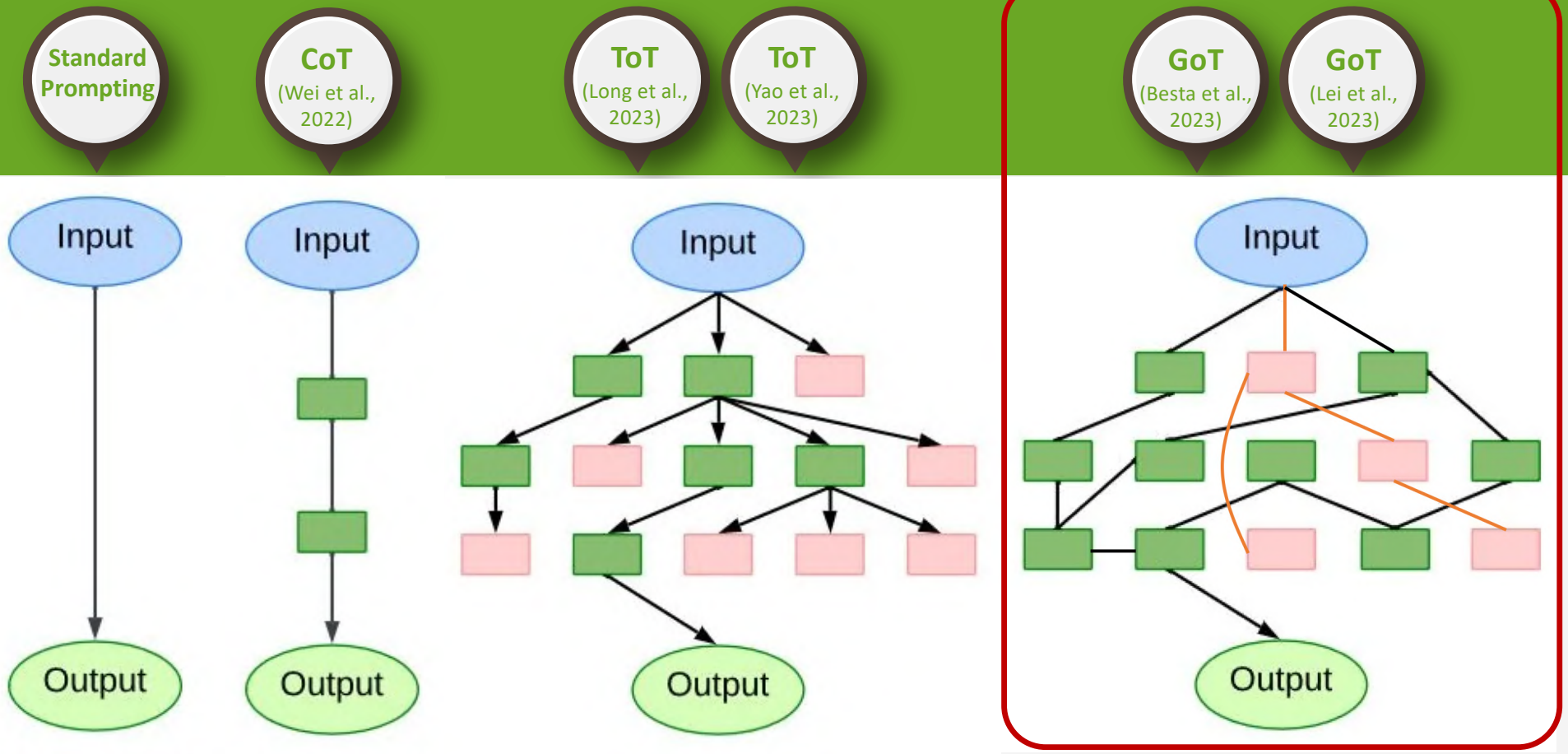
Volume: For a thought t, the number of preceding thoughts that could impact t



✗✗ **Low Volume: $O(\log_k N)$**

✓✓ **Low Latency: $O(\log_k N)$**

Prompt Engineering Techniques (Prompting Paradigms)



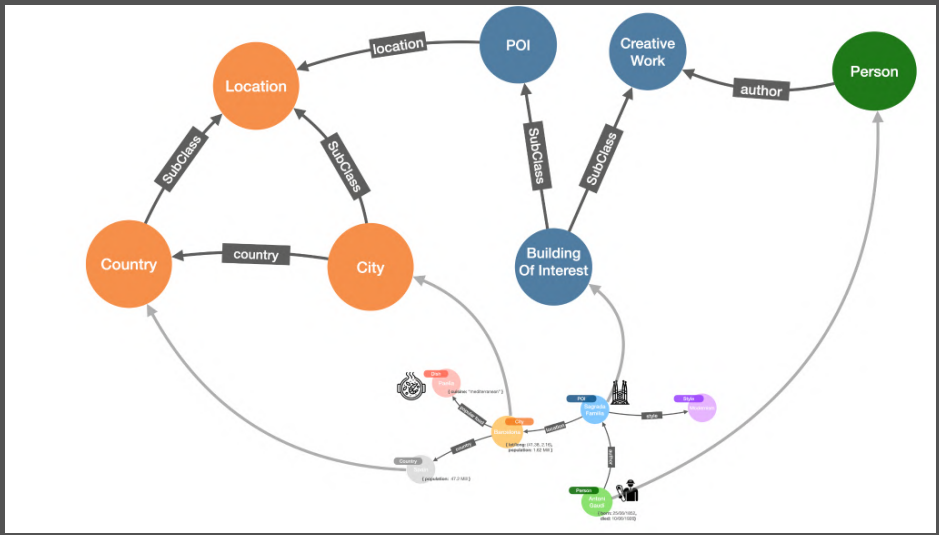
https://www.linkedin.com/posts/tonyseale_gpt4-promptengineering-semanticweb-activity-7075381524631580672-TAv3

04

Simulate Human Thinking

Graph-of-Thoughts

- Description of how a graph-of-thoughts works
- Concepts and mechanisms of graphs-of-thoughts
- Pros and cons of using graphs-of-thoughts



"…model the LLM reasoning as an arbitrary graph, where thoughts are vertices and dependencies between thoughts are edges" (Besta et al., 2023)

04

Simulate Human Thinking

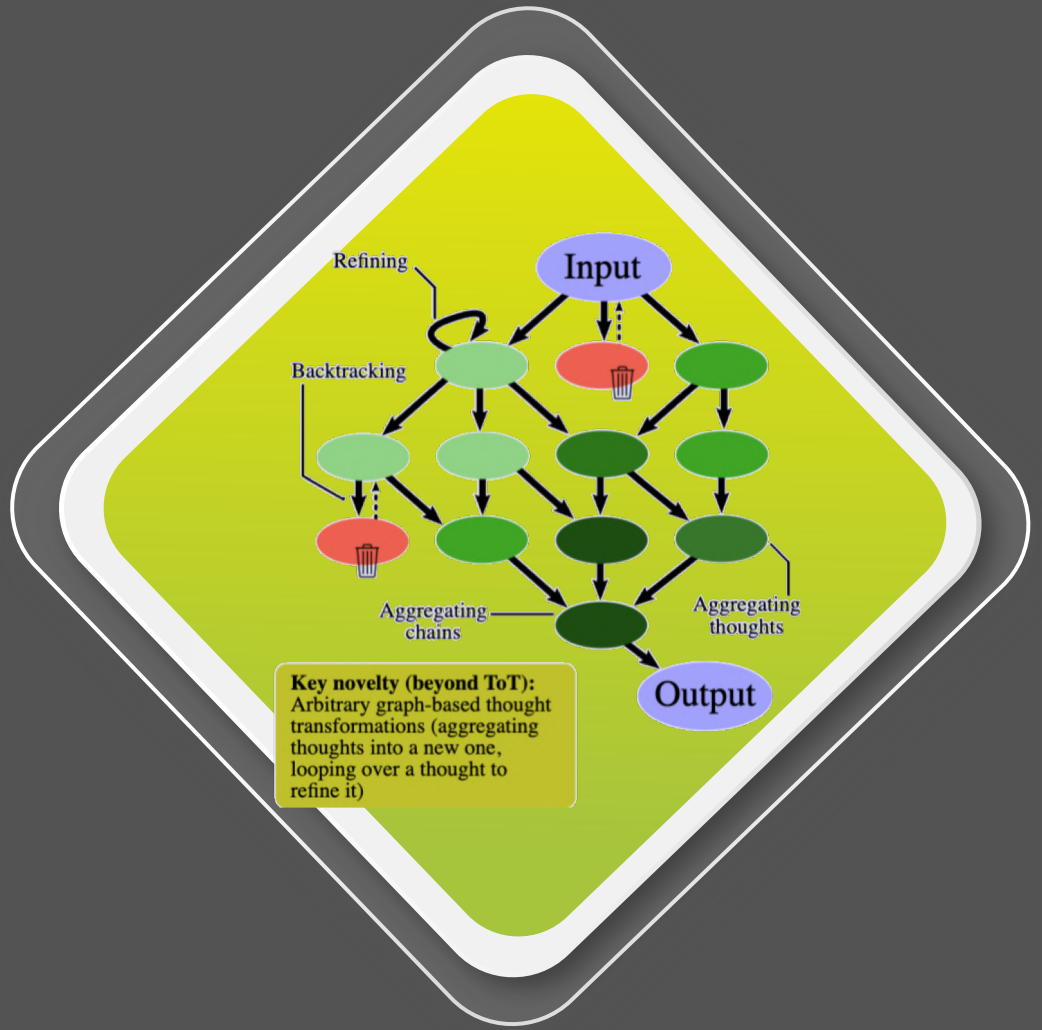
Graph-of-Thoughts

Prompter — Prepare the prompt to be sent to the LLM

Parser — Extract information from the LLM's generated thoughts

Validator — Verify the correctness conditions of a given LLM's thought

Controller — Implements a strategy for selecting thoughts from its Graph of Reasoning Structure (GRS)



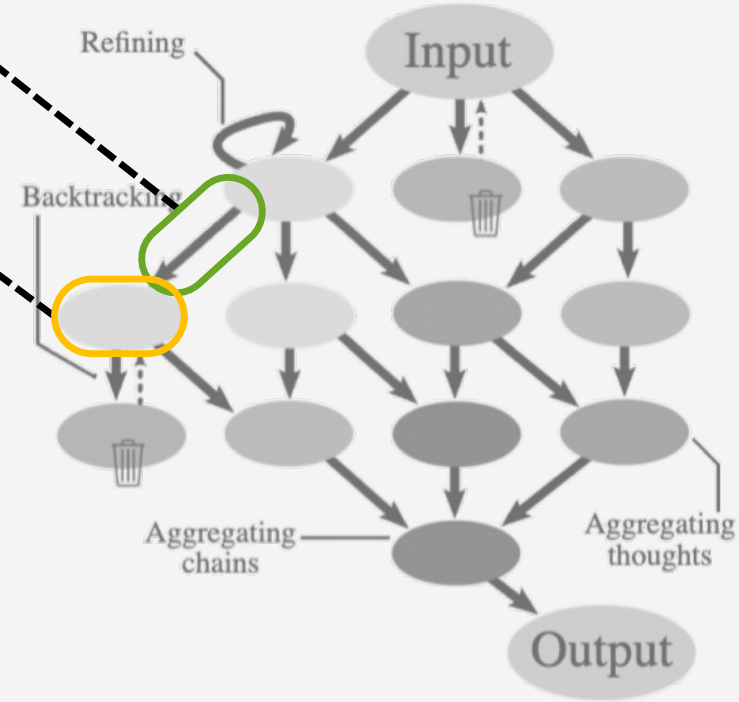
https://demogpt.medium.com/graph-of-thoughts-solving-elaborate-problems-with-large-language-models-78985558345

(Besta et al. 2023) https://arxiv.org/pdf/2308.09687v2

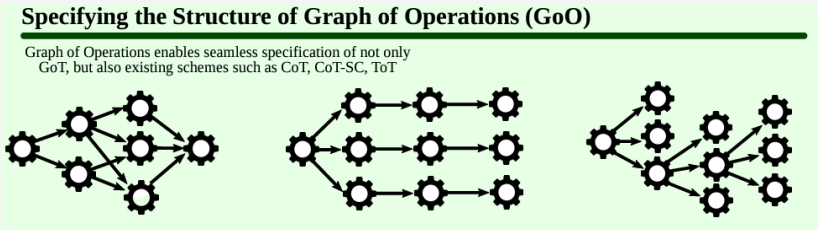
Graph-of-Thoughts: Graph-enabled “Transformations of Thought”

“Operation”:
Construction of thought with the previous thought as direct input

“Thought”:
Solution to a problem at-hand



“Graph of Operations (GoO)”:
A static structure that specifies the graph decomposition of a given task



Characteristics of GoT
- Directed Graph
- Can be heterogeneous

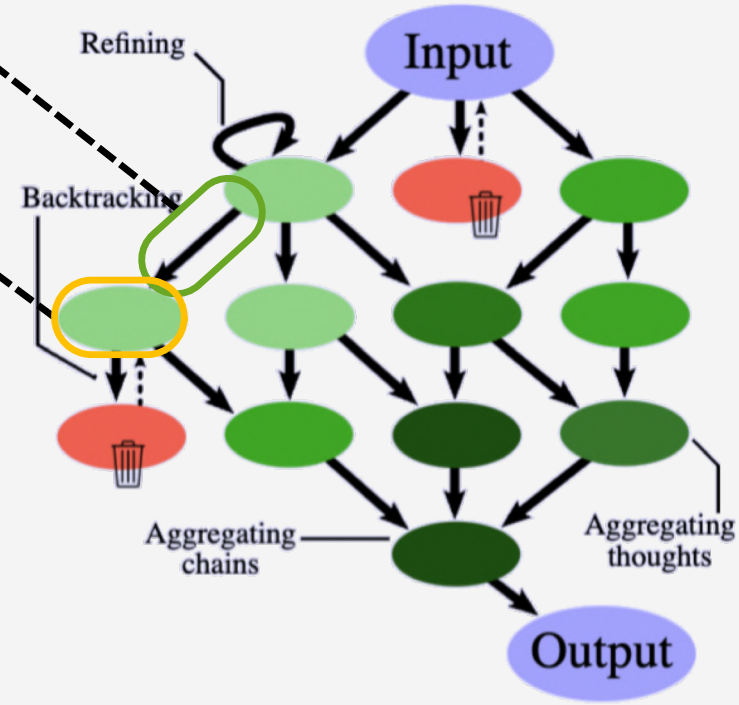
E.g., In creative writing, some vertices can be the writing plans while others can be the actual paragraphs of text

(Besta et al. 2023) https://arxiv.org/pdf/2308.09687v2

Graph-of-Thoughts: Graph-enabled "Transformations of Thought"

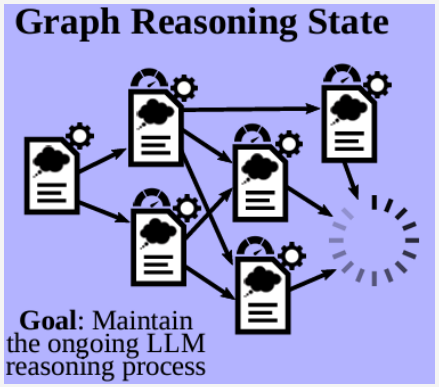
"Operation":
Construction of thought with the previous thought as direct input

"Thought":
Solution to a problem at-hand



"Graph of Operations (GoO)":
A static structure that specifies the graph decomposition of a given task

"Graph Reasoning State (GRS)":
A dynamic structure that maintains the current state of reasoning, e.g., states of individual thoughts (i.e., the intermediate outcomes of operations), their validity, and scores



Characteristics of GoT
- Directed Graph
- Can be heterogeneous

E.g., In creative writing, some vertices can be the writing plans while others can be the actual paragraphs of text

(Besta et al. 2023) https://arxiv.org/pdf/2308.09687v2

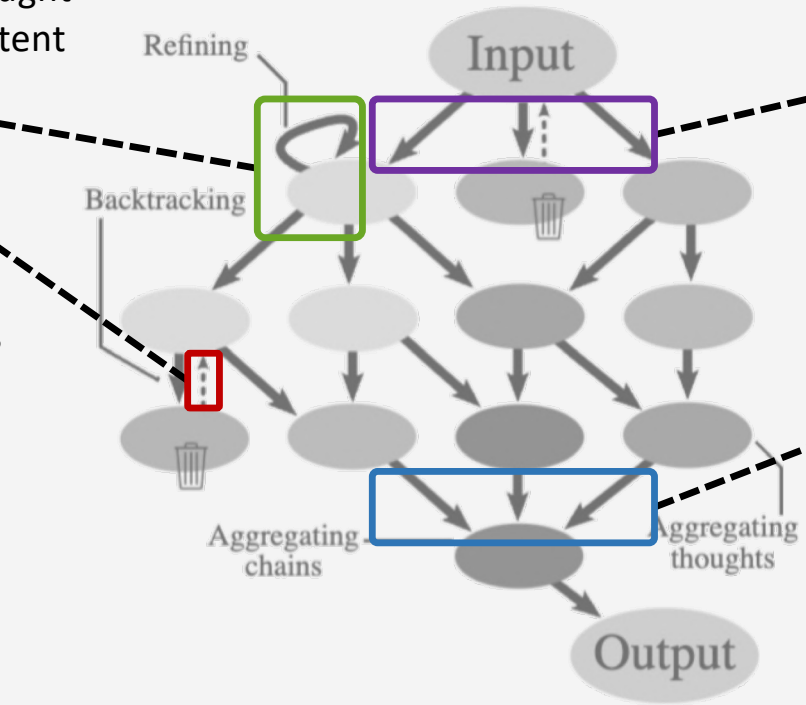
Graph-of-Thoughts: Graph-enabled "Transformations of Thought"

Refine a current thought by modifying its content

"Improve"

"Improve"

Backtrack from non-promising outcomes



Generate one or more new thoughts based on an existing single thought

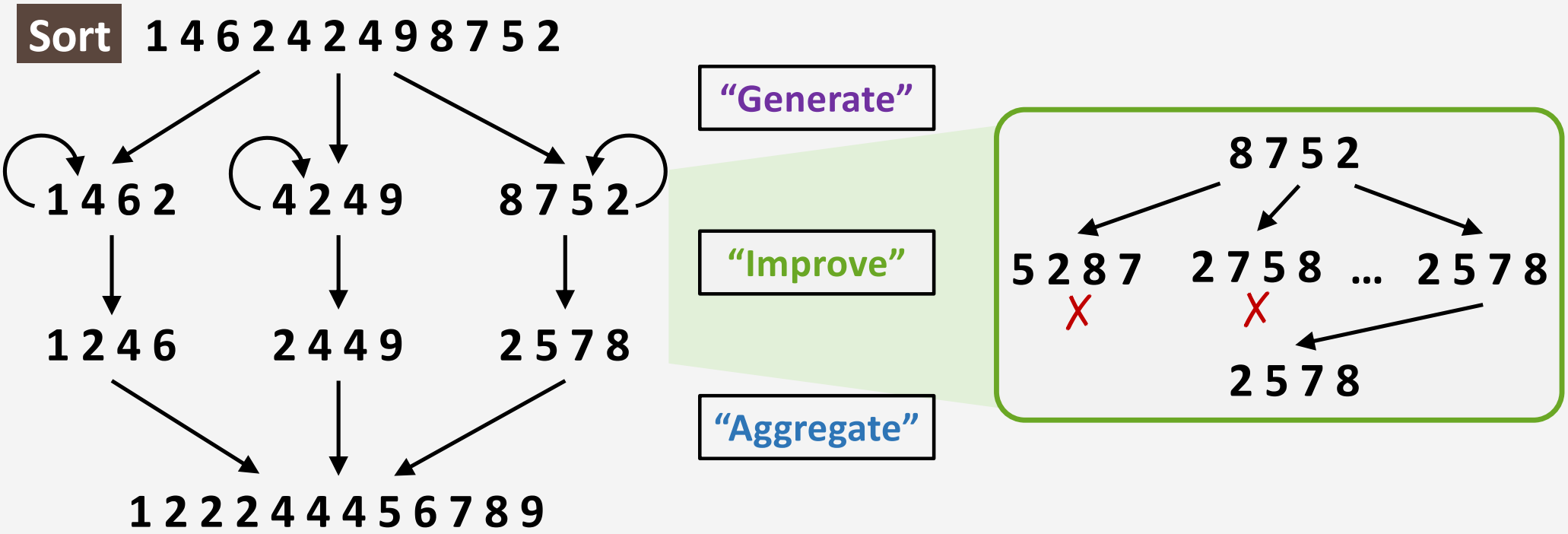
"Generate"

"Aggregate"

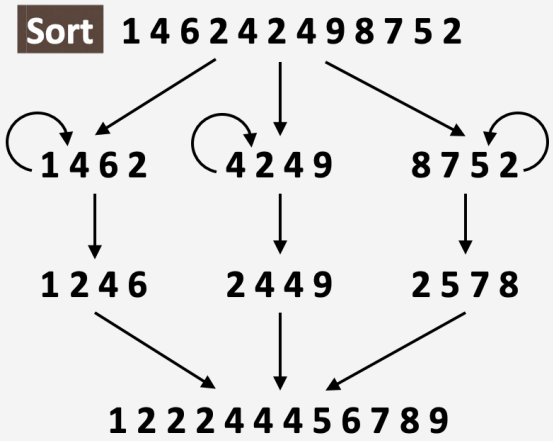
Aggregate arbitrary thoughts into new ones, to combine and reinforce the advantages of these thoughts, while eliminating their disadvantages.

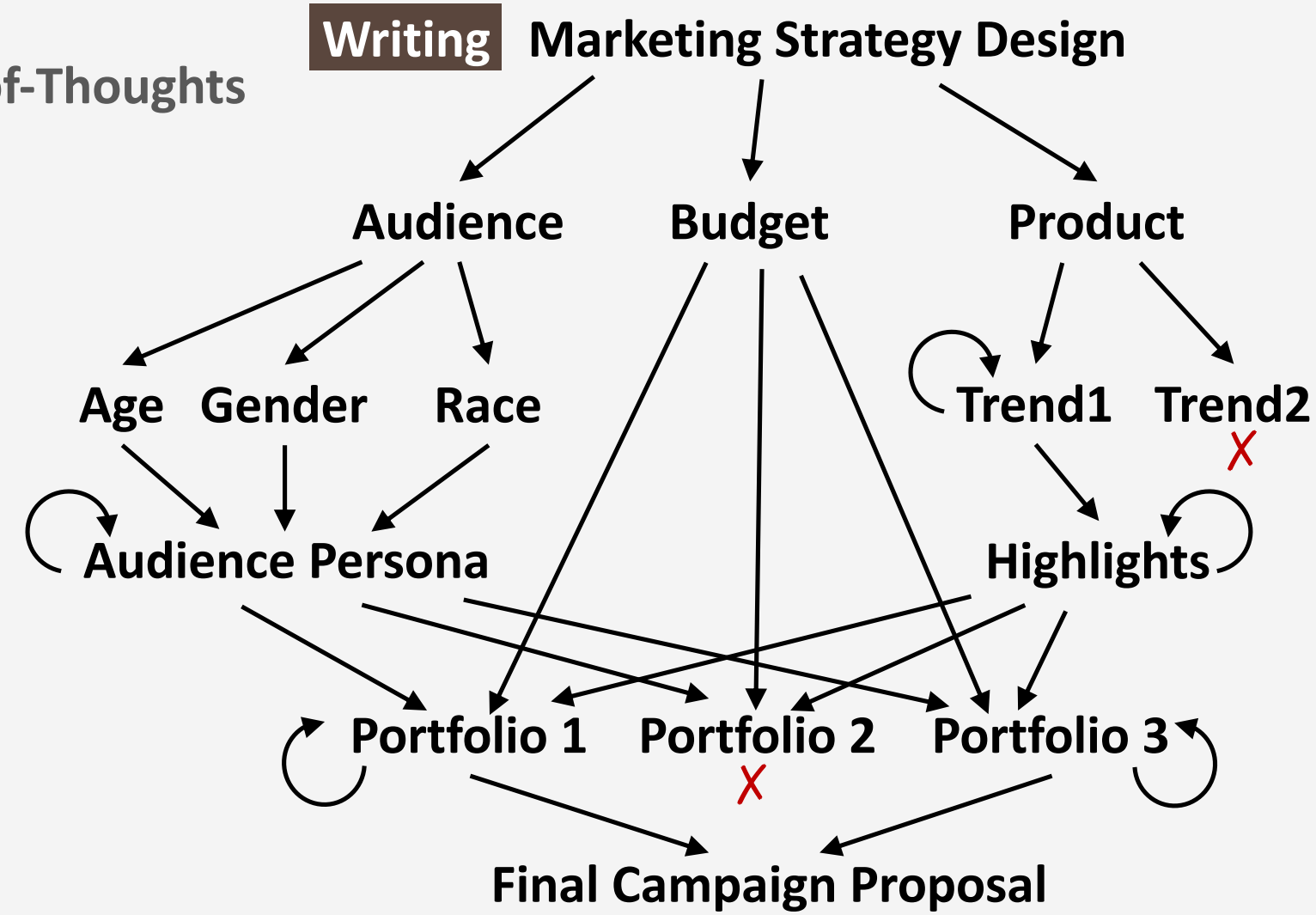
(Besta et al. 2023) https://arxiv.org/pdf/2308.09687v2

Examples of Graph-of-Thoughts

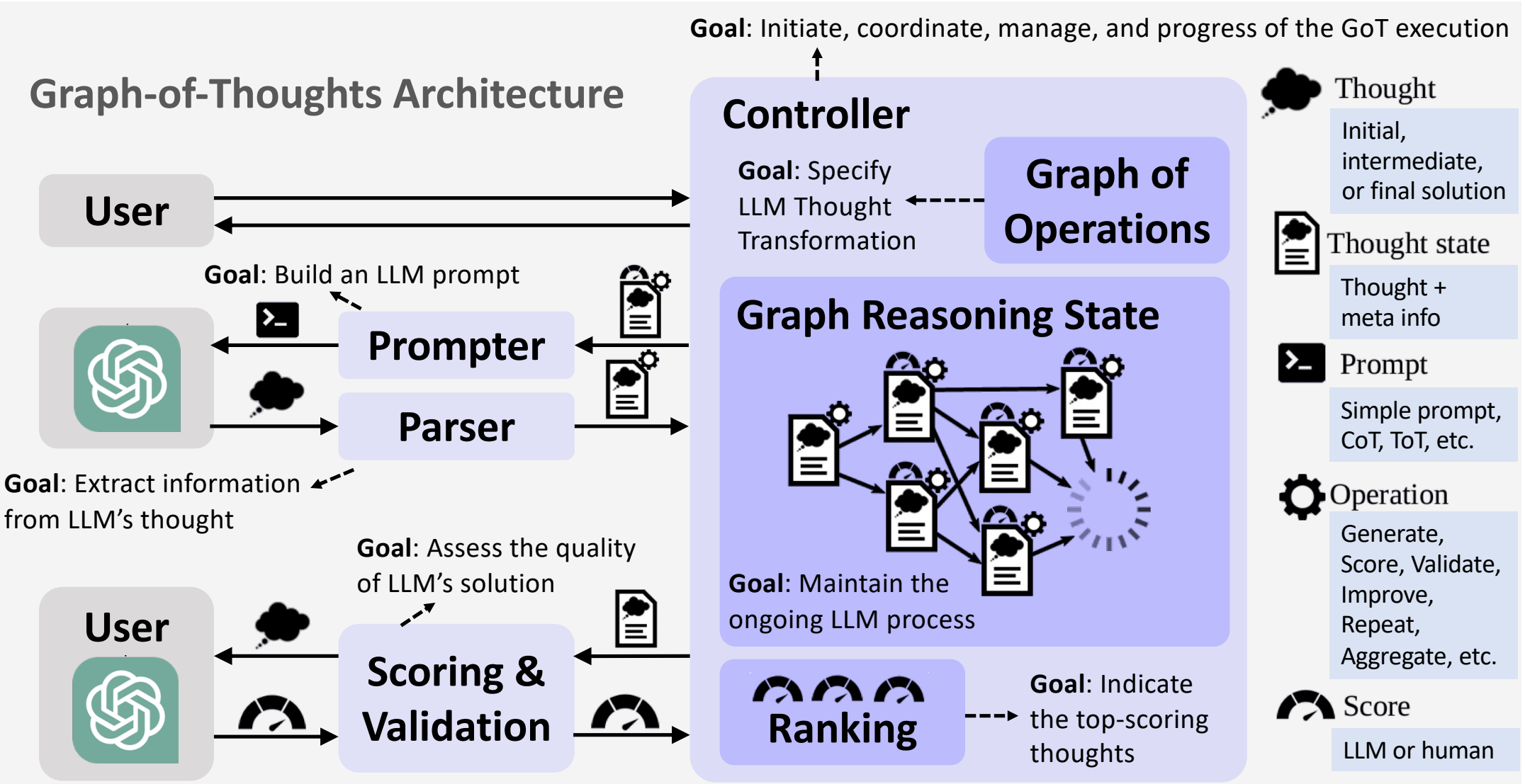


Examples of Graph-of-Thoughts

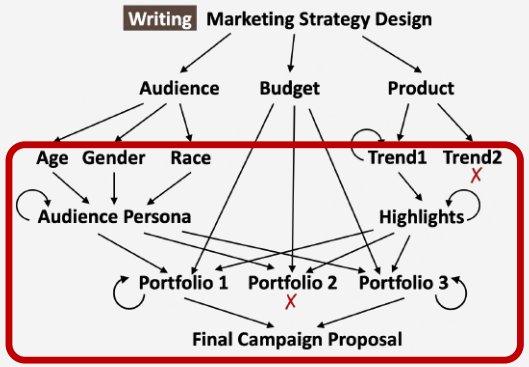




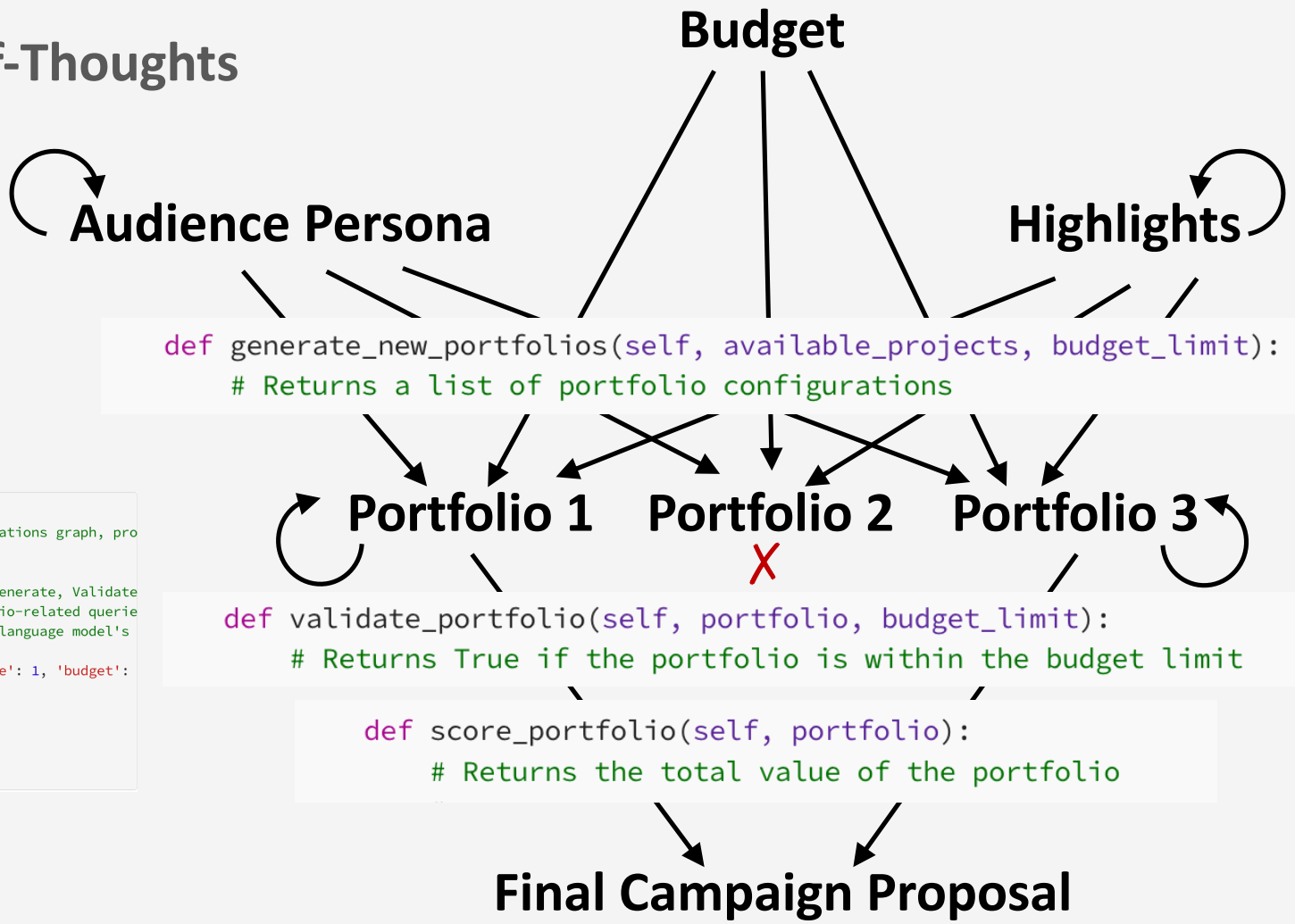
Graph-of-Thoughts Architecture



Examples of Graph-of-Thoughts

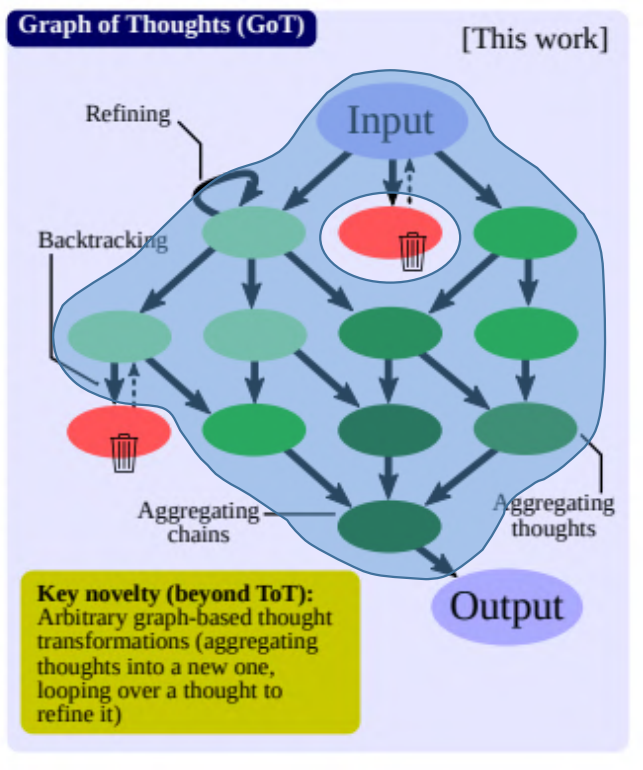



```
# Initialization of the Controller with the language model, operations graph, pro
controller = Controller(
    lm=ChatGPT(),  # Instance of the language model
    graph=graph_of_operations,  # Operations graph composed of Generate, Validate
    prompter=PortfolioPrompter(),  # Custom prompter for portfolio-related querie
    parser=PortfolioParser(),  # Custom parser to interpret the language model's
    problem_parameters={
        'available_projects': [{'value': 3, 'budget': 2}, {'value': 1, 'budget':
        'budget_limit': 2,
        'generated_portfolio': None
    }
)
```



Assuming a **fixed thought size** (#tokens) and a **fixed context size** ***N*** (#thoughts in the LLM context)

Latency: Number of steps between the input and output



Volume: For a thought t, the number of preceding thoughts that could impact t

✓✓ **Large Volume: *N***

✓✓ **Low Latency: $O(log_k N)$**

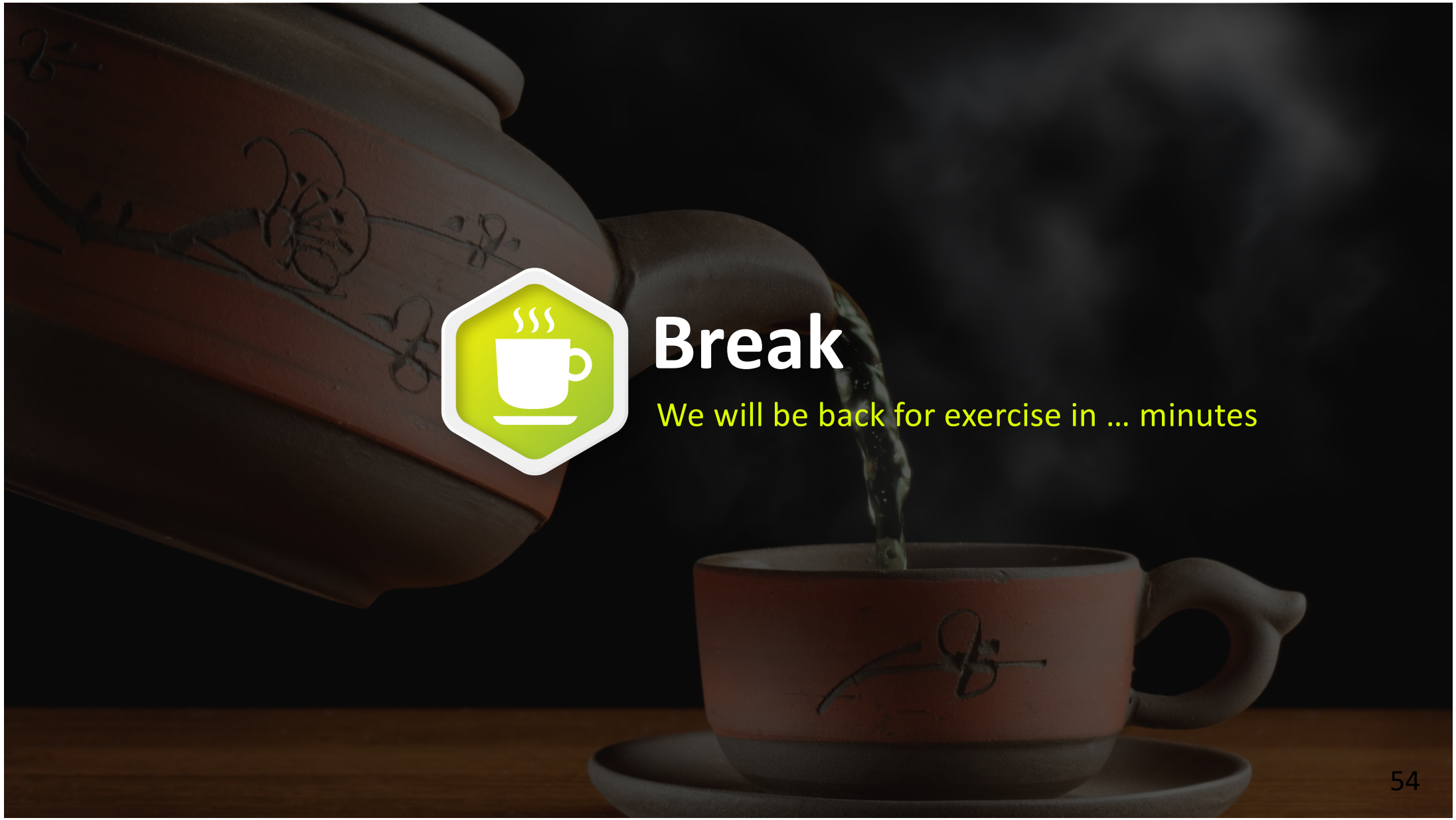
GoT Key Characteristics

- Suitable for elaborate problem cases
- Higher quality of outcome
- Higher cost

GoT Areas of Improvement

- Balance decomposition of tasks
  - Size of graph
  - Executability
  - Accuracy to avoid improve operation
- Reduce static prompt overhead
  - Difficulty in defining the GoO
  - Decrease the size of few-shot examples
  - …

https://www.youtube.com/watch?v=n10_cdSGjXY


Break
We will be back for exercise in … minutes

Applications of CoT, ToT, and GoT in Conversational Interactions

General Task Scenarios

- Arithmetic reasoning
- Common sense reasoning
- Content creation
- Data analysis
- Dialogue task
  - Give instruction
  - Collect info
  - Q&A
  - ...
- ...

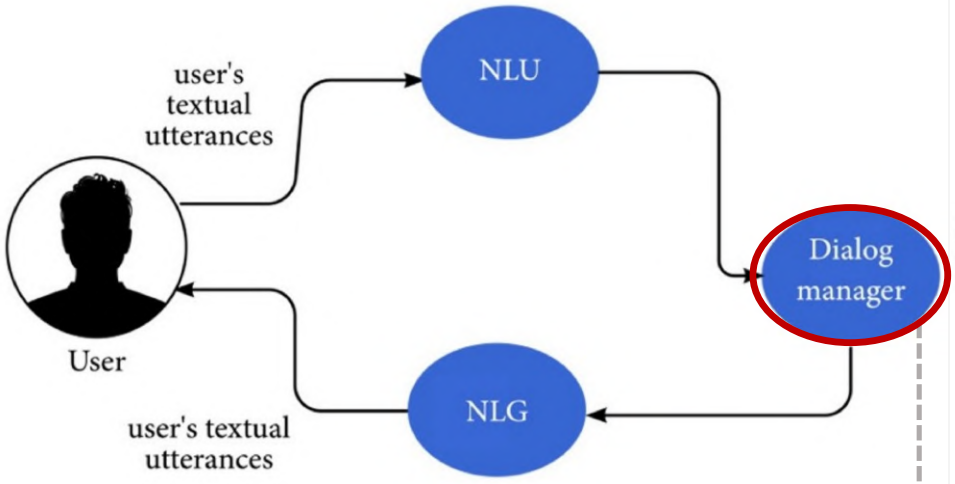
Dialogue Task Domains

Conversational Interactions

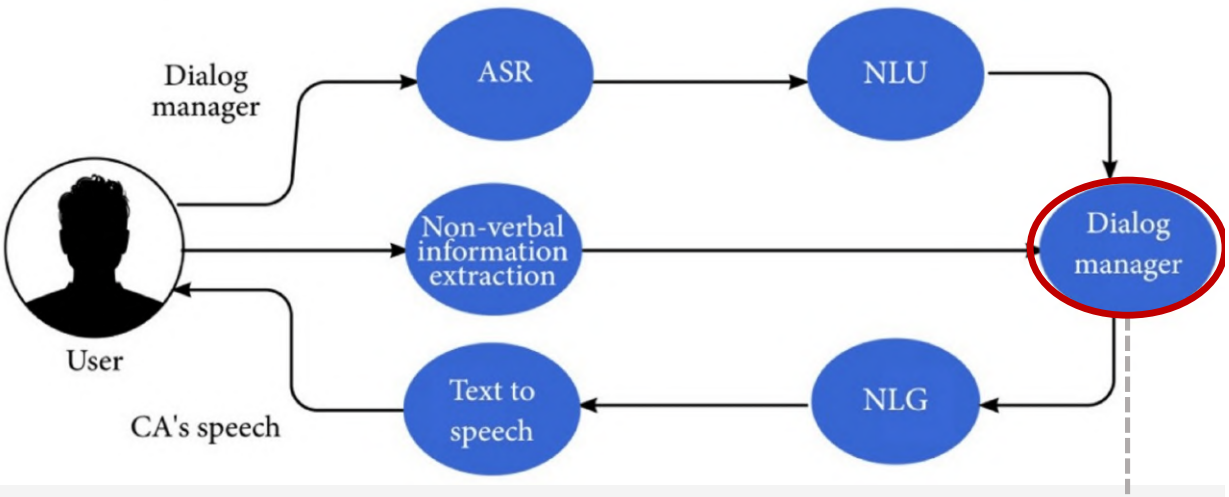
- Customer service
- Education (e.g., tutor)
- Healthcare (e.g., nurse)
- ...

Classic Architecture of a Conversational Agent (CA)

"Text-based CA"



"Speech-based CA"

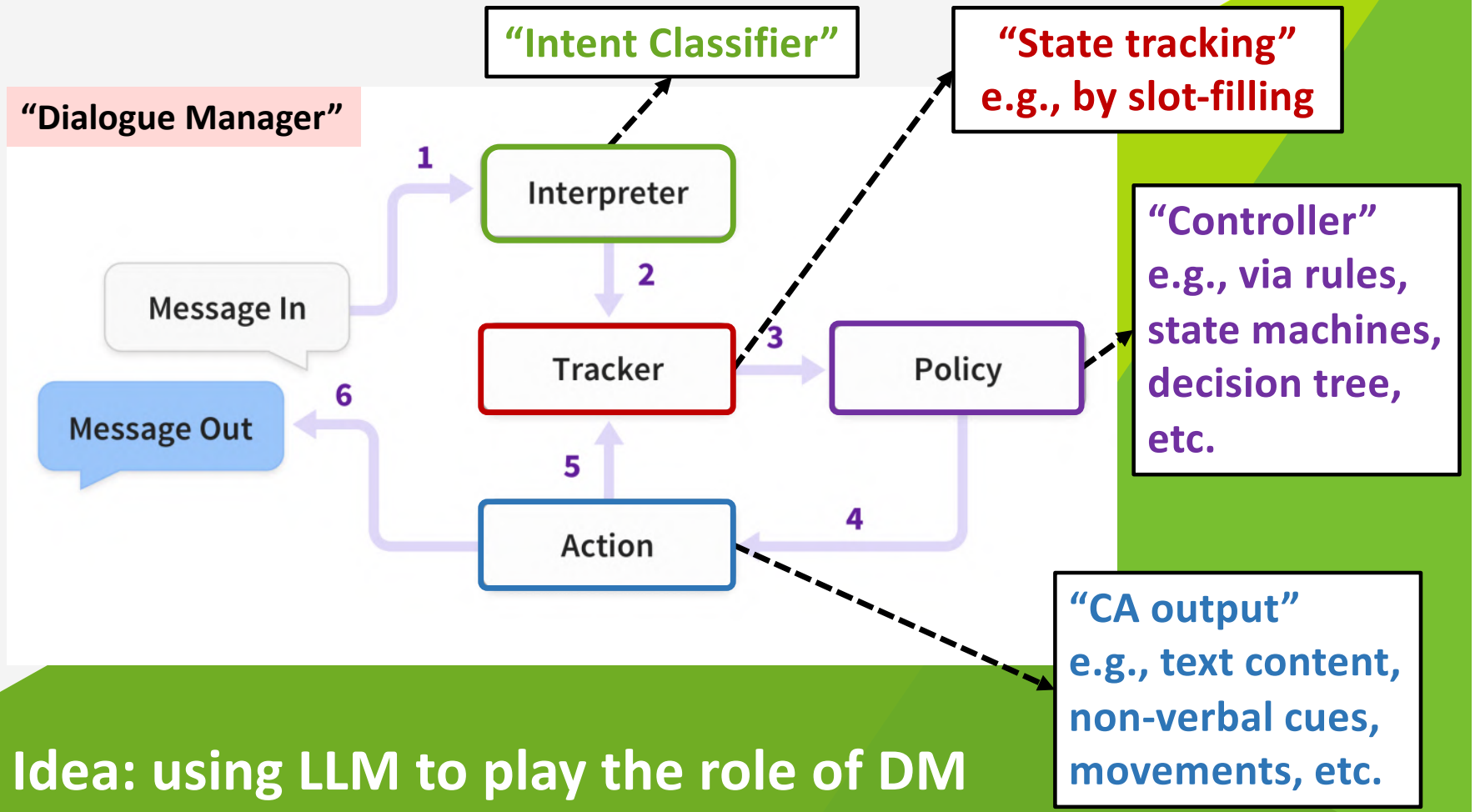


The aim of a Dialogue Manager (DM) is to mimic all cognitive aspects related to a natural conversation, handling dialogue states and flow.

https://www.mdpi.com/1424-8220/21/24/8448

A Close-up Look at a Dialogue Manager

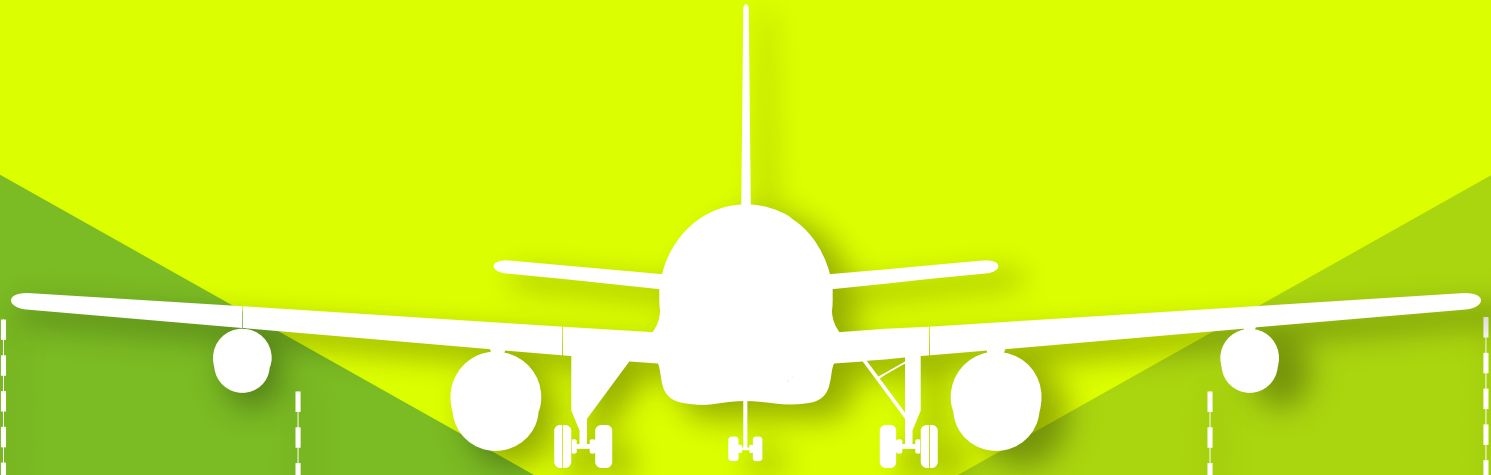
A Dialogue Manager interprets users' intents and dialogue context, keeps track of the current state of the conversation, selects the policy to proceed with, and controls the agent's output



Idea: using LLM to play the role of DM

https://paperswithcode.com/task/dialogue-management

Exercise



01 Introduction

02 Chain-of-Thought

03 Tree-of-Thoughts

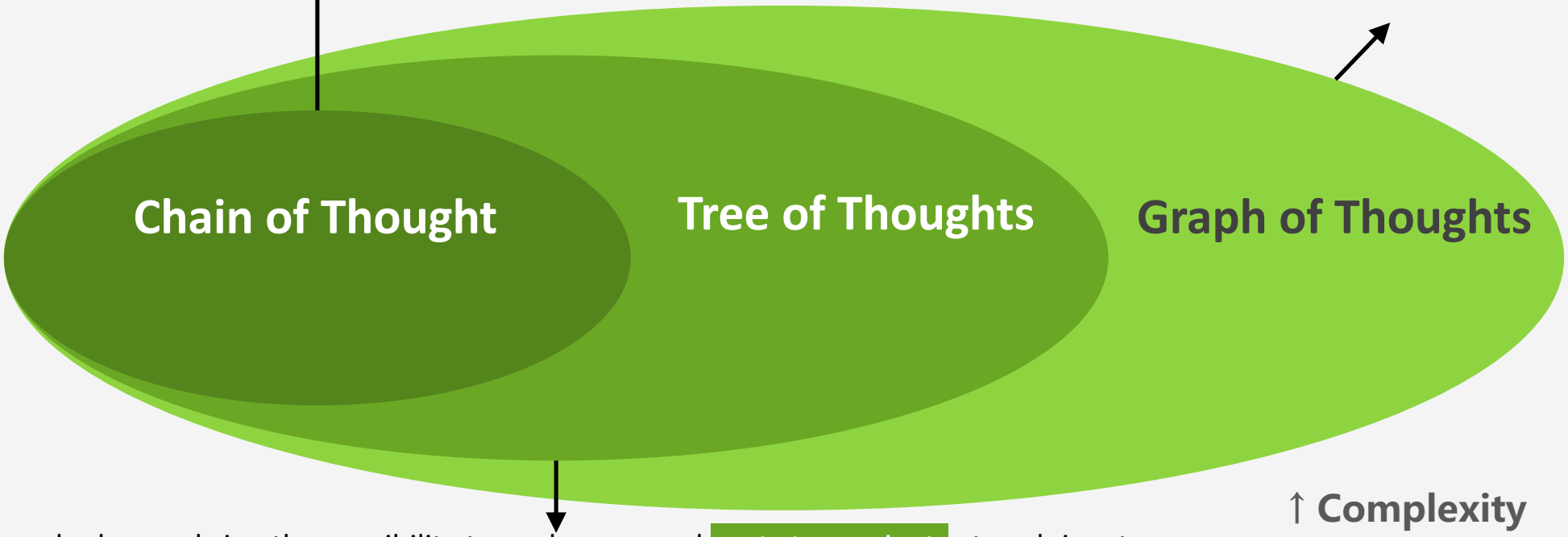
04 Graph-of-Thoughts

05 Applications

06 Exercise

Chain-based schemes introduce explicit intermediate LLM thoughts between the input and the output. This linear sequence of thoughts guides the LLM in a step-by-step manner towards the solution, enhancing the clarity and traceability of the reasoning process.

Graph-based schemes offer an arbitrary reasoning framework. They enable the aggregation of various reasoning steps into a synergistic solution, allowing for non-linear and multifaceted problem-solving approaches.



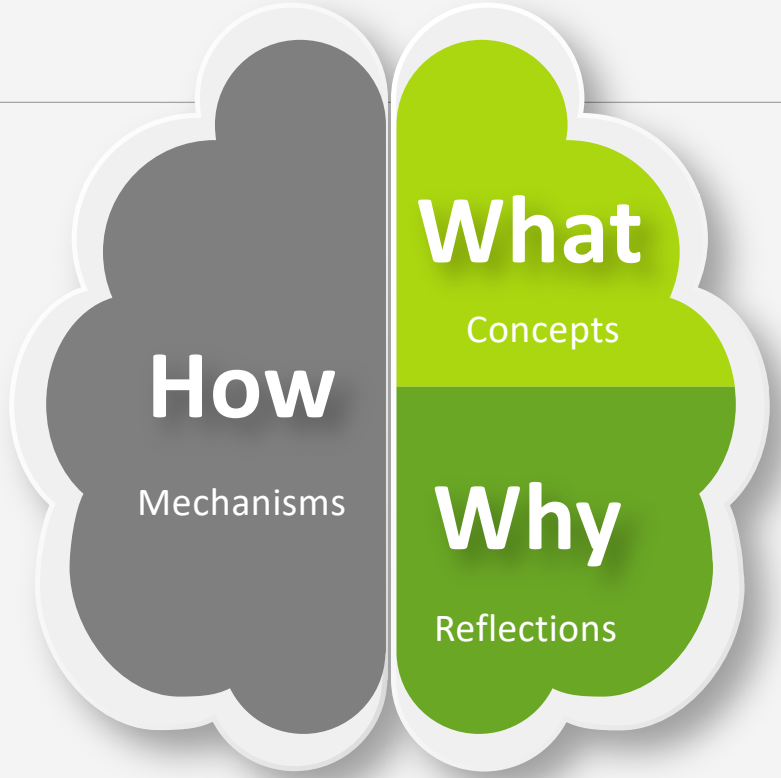
Tree-based schemes bring the possibility to explore several next-step variants at each juncture, allowing the LLM to evaluate multiple pathways and select the most promising one. This branching structure facilitates a broader exploration of potential solutions.

↑ Complexity
↑ Flexibility
↑ Cost and Load

Conclusion

Summary

- Fundamentals
  - Introduction of Chain-of-Thought, Tree-of-Thoughts, and Graph-of-Thoughts
- Examples
  - How to apply them in general and dialogue tasks
- Discussion
  - Pros and cons



Future Directions

- Improve cost-effectiveness,
  - E.g., encoding the structure within a single prompt
- Explore new architectures
  - Productivity
  - Programmability
  - Scalability
  - Parallelizability
- Develop understanding
  - Theoretical
  - Empirical

Charting the Routes of Thoughts in LLM-Empowered Conversational Interactions

Xiaojuan Ma

Human-Computer Interaction Initiative
Hong Kong University of Science and Technology
mxj@cse.ust.hk

Special Thanks: Dingdong Liu

CIX'24