

# Beyond Strawberry

gpt-o1

Info taken from:

<https://openai.com/index/learning-to-reason-with-llms/>

and my own illustrations / experiments

Presented by:

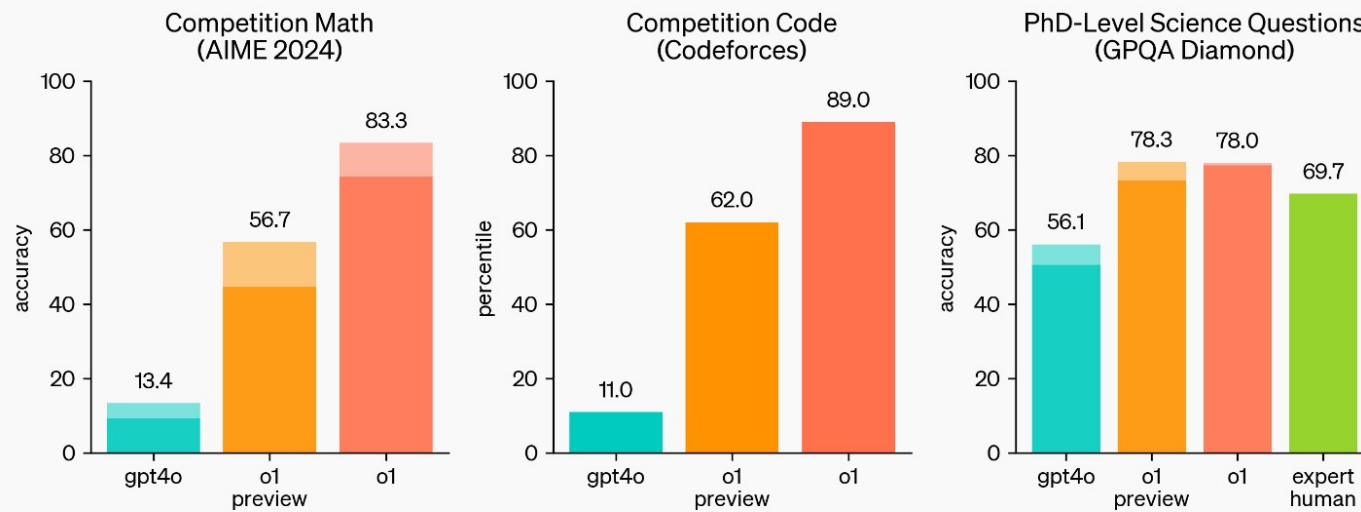
John Tan Chong Min

# Main Idea

- Instead of having the model output only once, why not iteratively do **Chain of Thought** so that the model can follow up on its earlier responses?
- Also, rather than just using only one sample output, why not use **multiple sample outputs** for diversified thinking and increased robustness, and then consolidate the result?

# Impressive Benchmarks

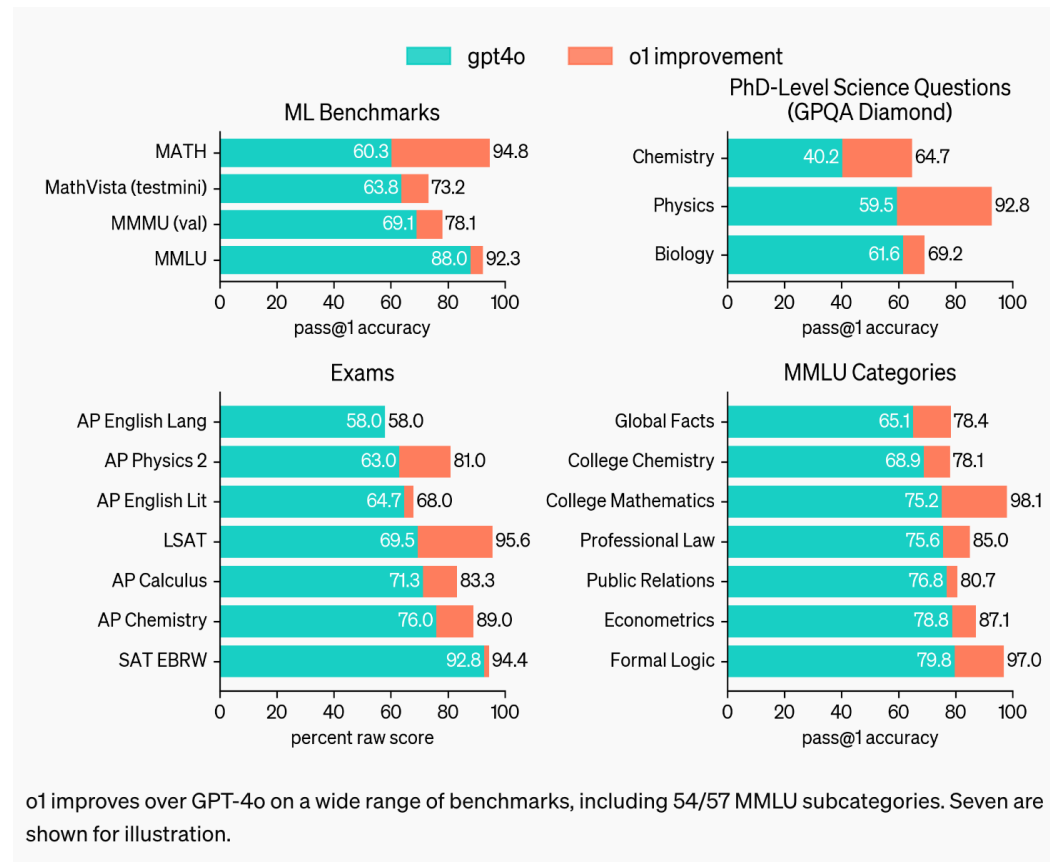
- Better performance in Math / Code / Science than gpt4o



o1 greatly improves over GPT-4o on challenging reasoning benchmarks. Solid bars show pass@1 accuracy and the shaded region shows the performance of majority vote (consensus) with 64 samples.

# Impressive Benchmarks

- Pretty significant gains on various examination benchmarks



# Speculation: Purely text action space without tool use

What are the action spaces available to you?



Thought for 10 seconds ▾

I can assist with a wide range of tasks that involve generating and understanding text, such as:

- **Answering questions** on various topics
- **Providing explanations** and clarifications
- **Assisting with coding problems** and debugging
- **Offering writing suggestions** and editing help
- **Translating text** between languages
- **Summarizing articles or documents**
- **Engaging in conversations** on numerous subjects

However, I operate entirely through text and do not have the ability to perform actions outside of generating responses.

Caveat - Is it only good performance on trained benchmarks?

#### ARC-AGI-PUB MODEL BASELINES

OpenAI `o1-preview` and `o1-mini` both outperform `GPT-4o` on the ARC-AGI public evaluation dataset. `o1-preview` is about on par with Anthropic's Claude 3.5 Sonnet in terms of accuracy but takes about 10X longer to achieve similar results to Sonnet.

Name	Score (public eval)	Verification Score (semi-private eval)	Avg Time/Task (mins)
o1-preview	21.2%	18%	4.2
Claude 3.5	21%	14%	0.3
o1-mini	12.8%	9.5%	3.0
GPT-4o	9%	5%	0.3
Gemini 1.5	8%	4.5%	1.1

<https://arcprize.org/blog/openai-o1-results-arc-prize>

The case of the strawberry

# Strawberry (gpt-4o)

- Wrong answer
- Tokenisation does not see letters individually, and counting is a cognitive task not done easily by LLMs

How many r are there in strawberry?



There are 2 "r"s in the word "strawberry."



# Strawberry (gpt-o1)

- Perhaps breaking down into steps helps?
- Perhaps this is in training dataset

How many r are there in strawberry?



Thought for 4 seconds ^

## Counting letter occurrences

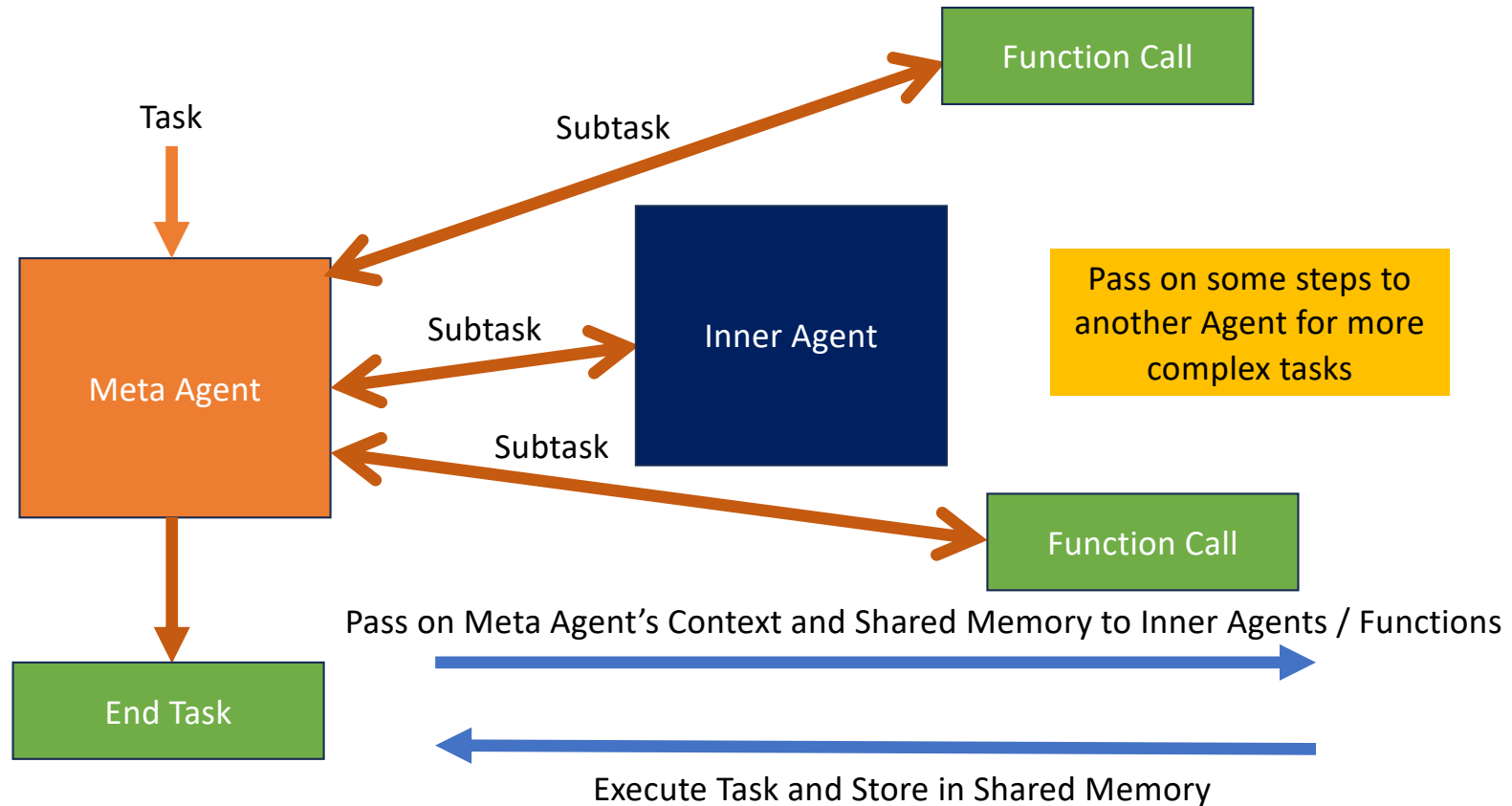
I'm mapping out how to count the number of 'r's in "strawberry." This involves listing the word and tallying each 'r' to reach a total. This method ensures a clear and accurate count.

## Counting occurrences

I'm counting the 'r' positions in the word "strawberry" and finding three 'r's at positions 3, 8, and 9.

There are three "r"s in the word "strawberry".

# TaskGen Overview



<https://github.com/simbianai/taskgen>

# Strawberry (TaskGen)

- Is gpt-o1 just wrapping gpt-4o inside an agentic framework?

```
agent = Agent('Generalist Agent',  
             '''You are to break down the task into smaller steps and critically reason over them.''' ,  
             llm = llm)
```

```
output = agent.run('How many r are there in strawberry')
```

**Observation:** No subtasks have been completed yet, and the task is to determine the number of occurrences of the letter "r" in the word "strawberry".

**Thoughts:** To complete the task, I need to analyze the word "strawberry" and count how many times the letter "r" appears in it.

**Subtask identified:** Count the occurrences of the letter "r" in the word "strawberry".

Getting LLM to perform the following task: Count the occurrences of the letter "r" in the word "strawberry".

> To count the occurrences of the letter "r" in the word "strawberry", I will first analyze the word itself. The word "strawberry" consists of the following letters: s, t, r, a, w, b, e, r, r, y. Next, I will identify and count each instance of the letter "r". In this case, the letter "r" appears three times in the word "strawberry". Therefore, the final count of the occurrences of the letter "r" is 3.

**Observation:** The occurrences of the letter "r" in the word "strawberry" have been counted and found to be three.

**Thoughts:** Since the task of counting the occurrences of "r" has been completed, the next step is to finalize the output and present it to the user.

**Subtask identified:** End Task

Task completed successfully!

```
# display inner CoT in a controlled manner
```

```
output = agent.reply_user('Summarise each subtask and display to the user in a concise manner. Format: Subtask: <subtask>\nSummary: <summary>', stateful = False)
```

Subtask: Count the occurrences of the letter "r" in the word "strawberry".

Summary: The task involved analyzing the word "strawberry" to identify and count the instances of the letter "r". Upon examination, it was determined that the letter "r" appears three times in the word. Therefore, the final count of the occurrences of the letter "r" is 3.

```
# reply user
```

```
output = agent.reply_user()
```

To determine how many times the letter "r" appears in the word "strawberry", I analyzed the word itself. The letters in "strawberry" are: s, t, r, a, w, b, e, r, r, y. Upon counting, I found that the letter "r" appears three times in total. Therefore, the final count of the occurrences of the letter "r" in "strawberry" is 3.

Possible “Thinking” Architectures

# “Traditional” Chain-of-Thought

## 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?

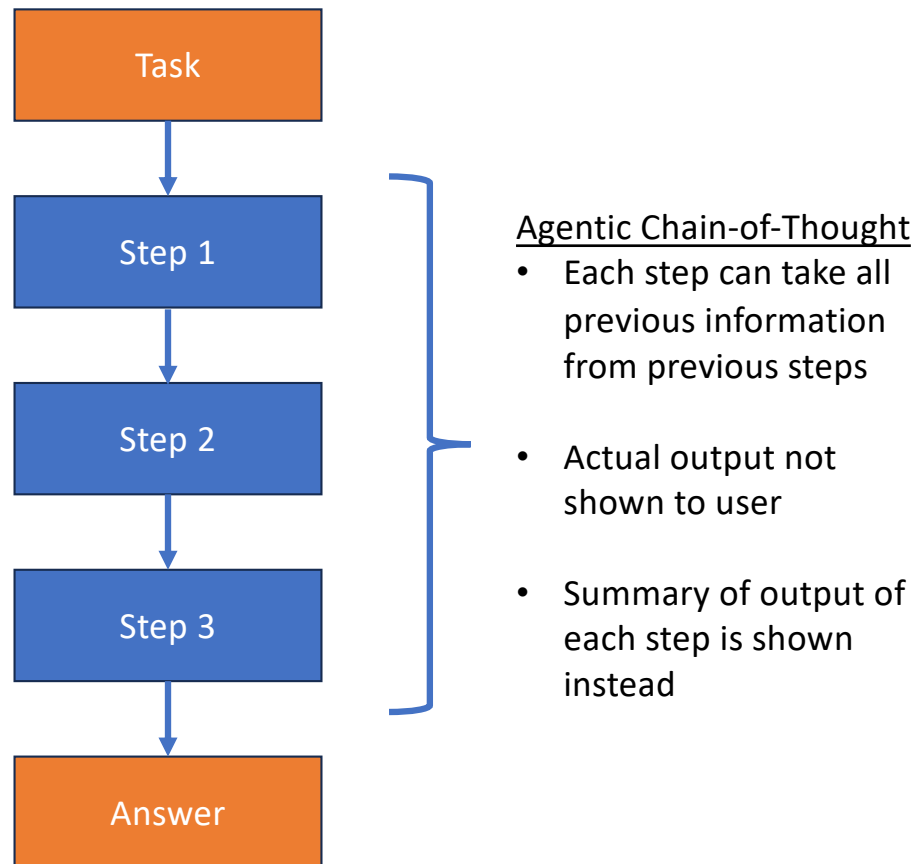
### Model Output

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. ✅

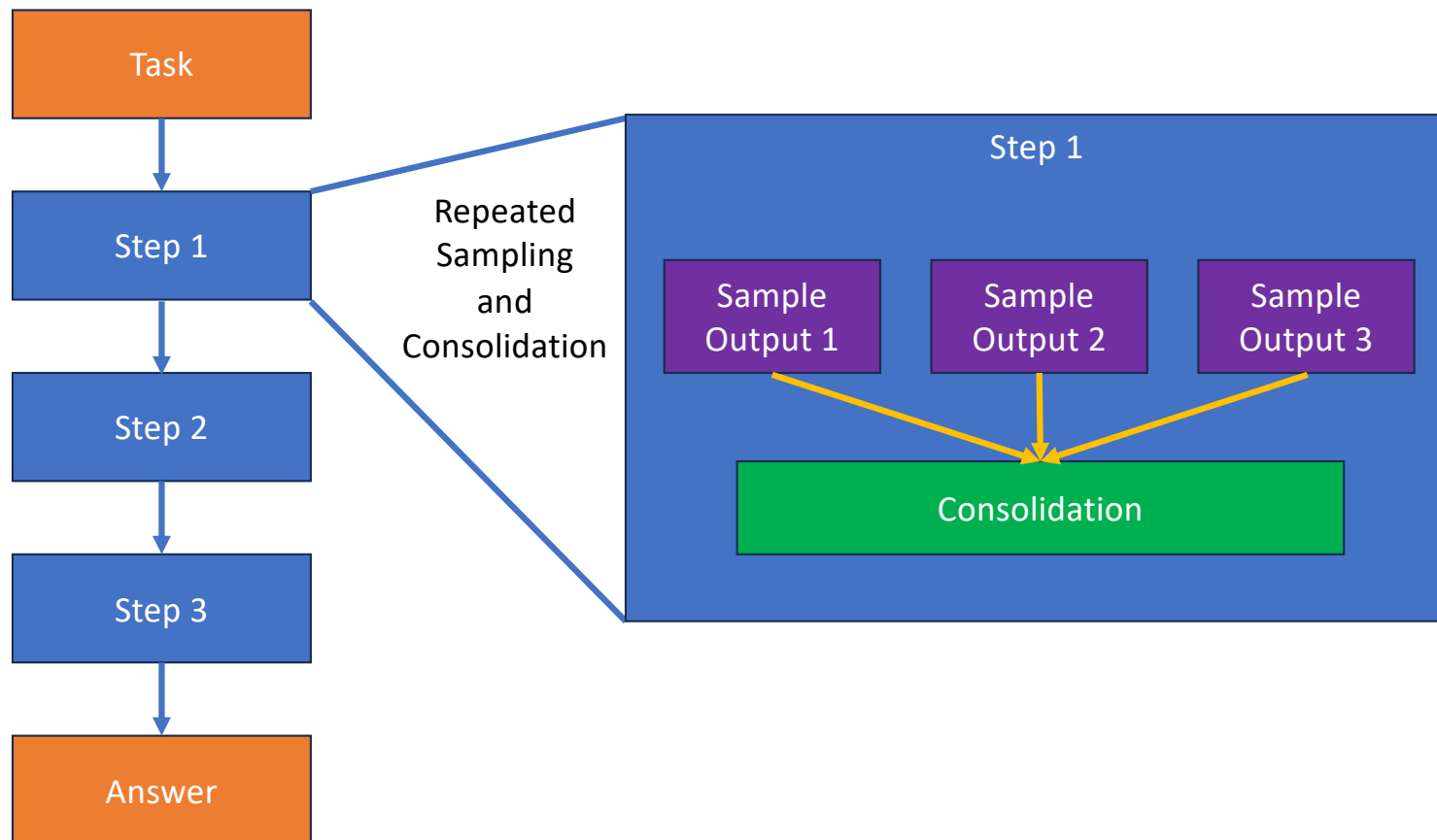
- Trained with more elaboration of reasoning steps
- Can also be zero-shot with “Let’s think step by step”  
(<https://arxiv.org/pdf/2205.11916>)
- Can also be guided via structured output (e.g. ReAct framework)  
(<https://arxiv.org/pdf/2210.03629>)

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. Jason et. al. 2022.

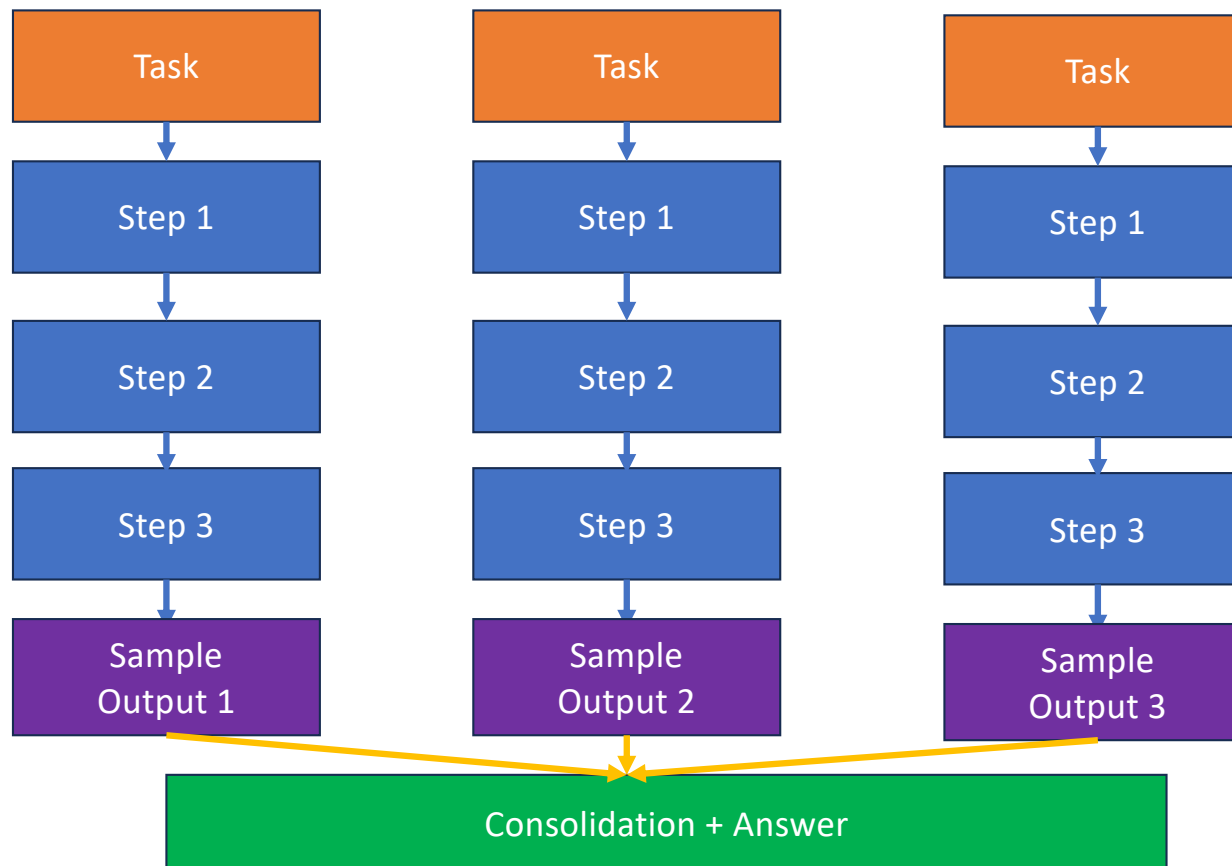
# Agentic Chain of Thought



# Multiple sampling and consolidation (Task-level)

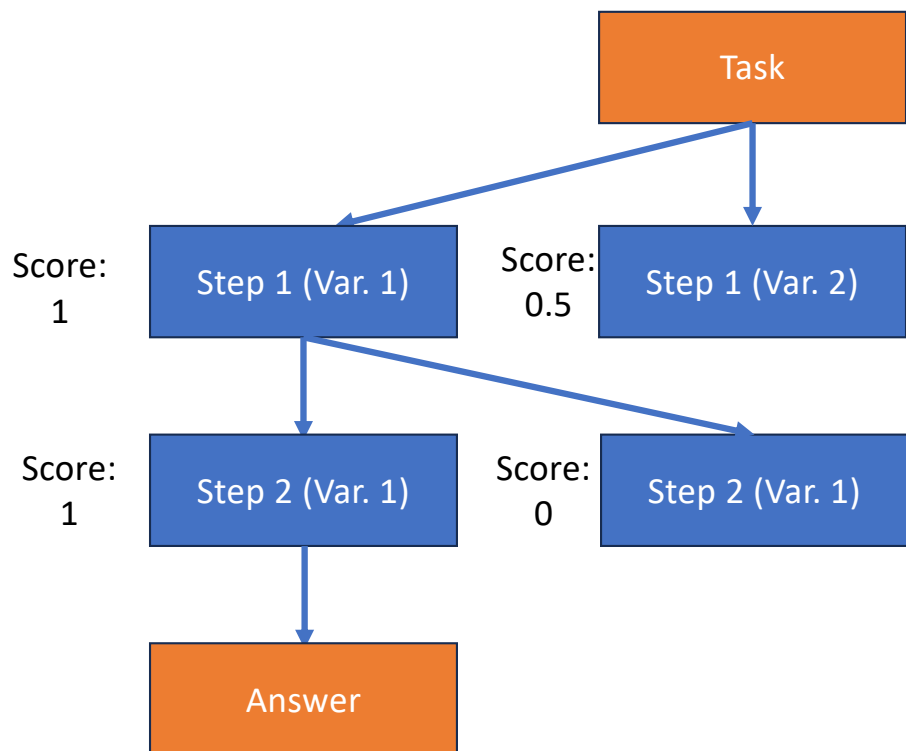


# Multiple sampling and consolidation (Agent-level)





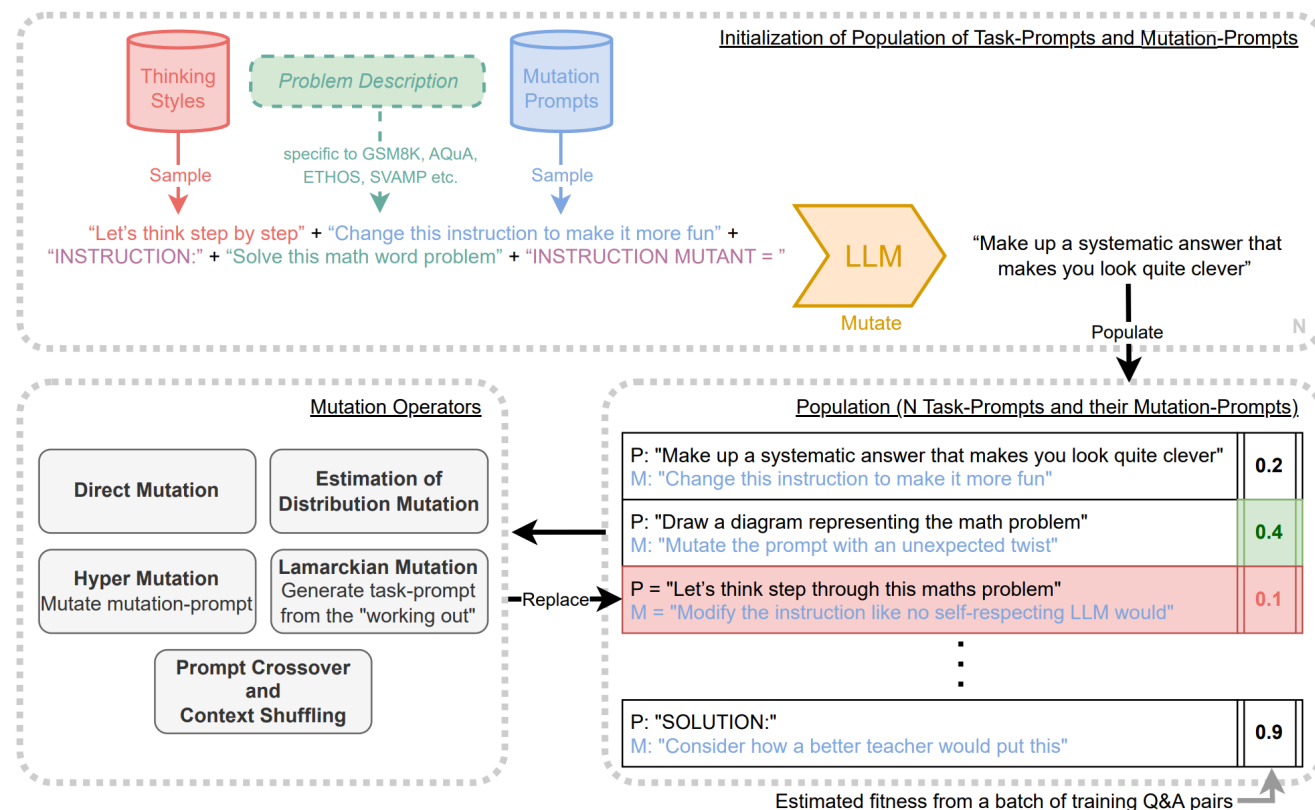
# Multiple sampling and consolidation (Tree Search)



- Use MCTS-like approach to navigate through the tree of steps
- At each point, pick a node that is most promising and expand it
- Evaluate each node with closeness of reaching the answer
  - May not be accurate
- Take top-k paths and consolidate them
- Similar ideas:
  - MCTS-r (<https://arxiv.org/abs/2406.07394>)
  - Tree of Thoughts (<https://arxiv.org/pdf/2305.10601>)

# How to ensure diversity in sample outputs?

- Perhaps it uses Mutation Prompts / hints to get it to change style of thinking
- If multi-agent system, could be different personalities of agents



PromptBreeder. 2023. DeepMind.

# Likely 2 repeated samples in web gpt-o1

How many r are there in strawberry?



Thought for 4 seconds ^

## Counting letter occurrences

I'm mapping out how to count the number of 'r's in "strawberry." This involves listing the word and tallying each 'r' to reach a total. This method ensures a clear and accurate count.

## Counting occurrences

I'm counting the 'r' positions in the word "strawberry" and finding three 'r's at positions 3, 8, and 9.

There are three "r"s in the word "strawberry".

without using chain of thought, what is the number of lillies in a pond in day 10 if day 1 has 1 lily, and it doubles every day?



Thought for 5 seconds ^

## Calculating lilies

I'm figuring out the number of lilies doubling from 1 on day 1. Each day, it doubles, reaching 10 lilies on day 10.

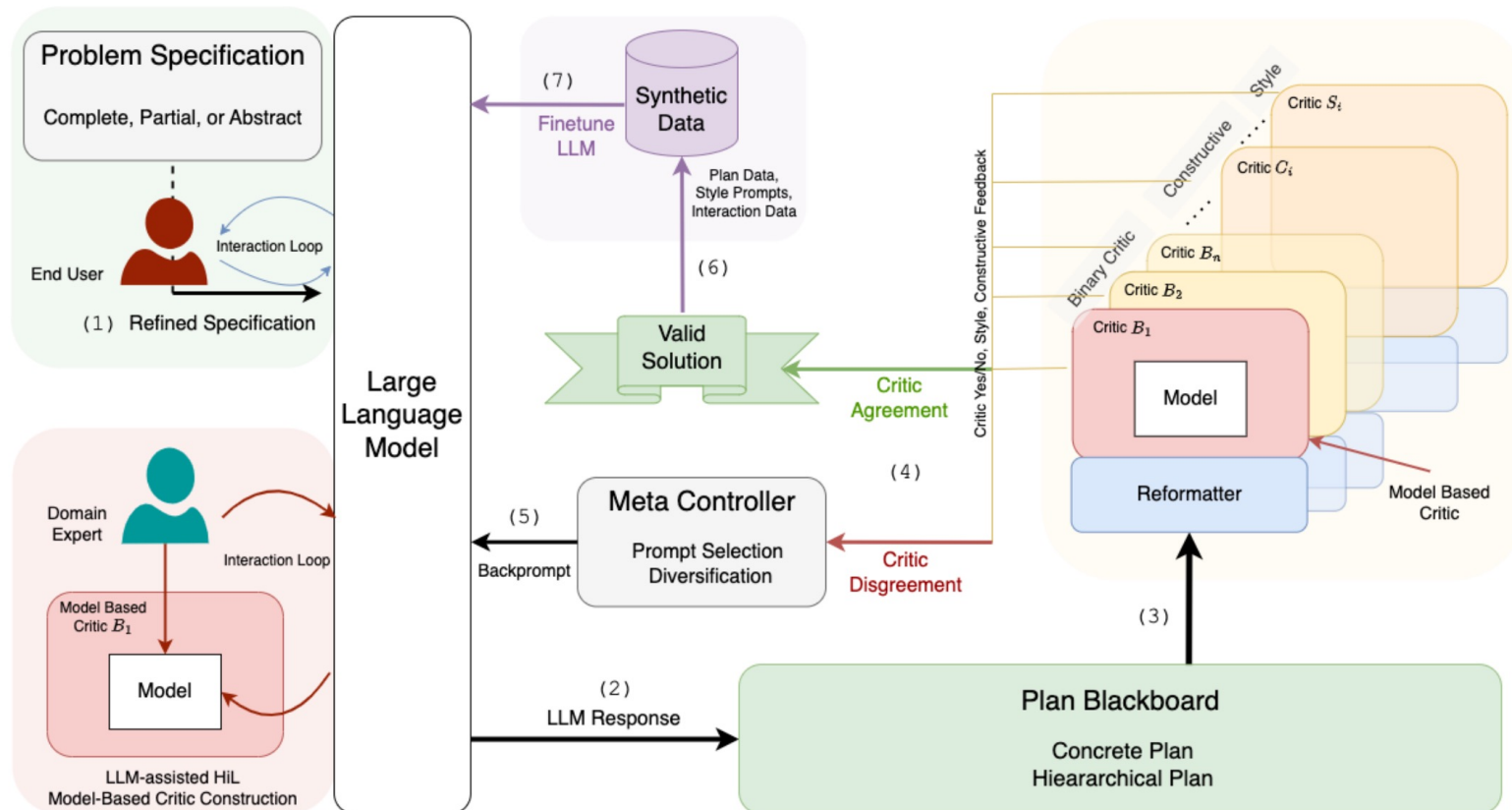
## Calculating lilies

OK, let me see: The number of lilies on day 10 is determined by multiplying 1 by 2 raised to the power of (10-1), resulting in a total of 512 lilies.

# Can LLMs be their own verifier / critic?

- LLMs cannot critique well if LLM is not an expert in that domain
- LLMs cannot critique well if it involves an abstraction space not semantically meaningful
  - E.g. Math
  - E.g. counting letters, such as 'r' in strawberry
- Best if interfaced with something that is rule-based and is known to work well, e.g. code generator and compiler, calculator

# LLM Modulo – Using external verifier/critics



LLM Modulo. 2024. Subbarao et. al.

My thoughts

# My thoughts (Part 1)

- gpt-o1 is an impressive model that can answer some questions previously answered wrongly by gpt-4o
- (Speculated) Lack of tool integration means that external suitable verifiers are not able to be used to improve LLM performance
- OpenAI should **package agentic structure over an LLM as a separate product**, rather than put it as an LLM with a unmodifiable pseudo-agentic structure
  - Greater compositionality and customisability
  - Model improvements can be directly compared, rather than packaging improvements as part of “agentic system”

# My thoughts (Part 2)

- Using Reinforcement Learning to train the LLM's thinking process is very inefficient
  - Difficult to compress task-following / result correctness with just a single scalar number
  - How is the reward model going to learn effectively across tasks?
- Multiple agents with different biases should be used instead to create diversified thinking processes
- (Speculation) gpt-o1 will likely fail at new problems requiring simulators (e.g. new version of tic tac toe)
- **Memory structures for learning and reasoning** is a huge missing part from gpt-o1



# Questions to Ponder

- Is Chain of Thought the best way to elicit reasoning?
- What are the potential downsides of reasoning purely via next-token prediction without any external structures / simulators?
- Are there any drawbacks of **thinking too much**?
- How could the dataset be curated for gpt-o1?
- Have we already reached the limit of what LLMs can do for problem solving? What is the next frontier?