

AI Agents

LLM-based agents



Ahmed Abdou

ahmed.abdou@tum.de

03.09.2024

Data Science Journal Club



Agents

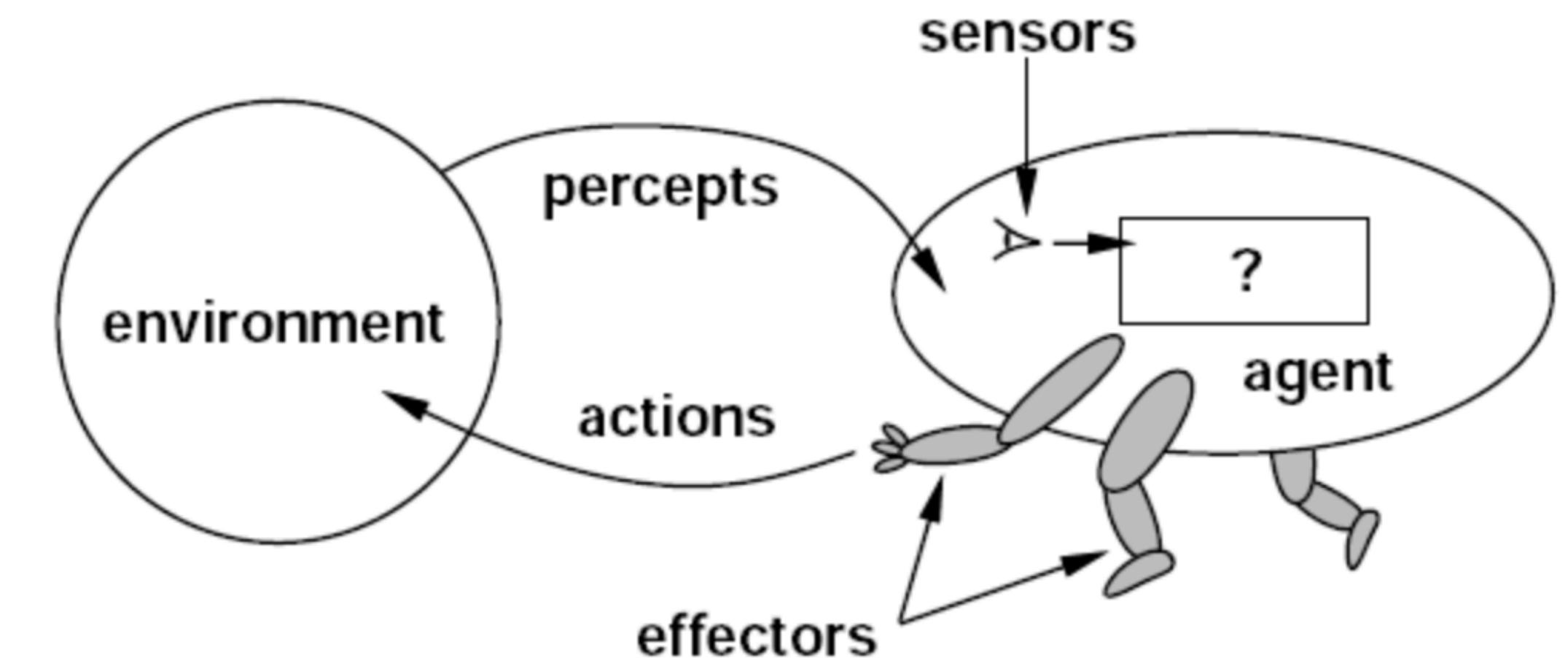
In philosophy

- an “agent” is an entity with the capacity to act.
- At the core of the standard conception are the following two claims:
 - action is to be explained in terms of the intentionality
 - there is a close connection between intentional action and acting for a reason

Agents In artificial intelligence

- "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors"

(Artificial Intelligence: A Modern Approach by Stuart Russell and Peter Norvig)



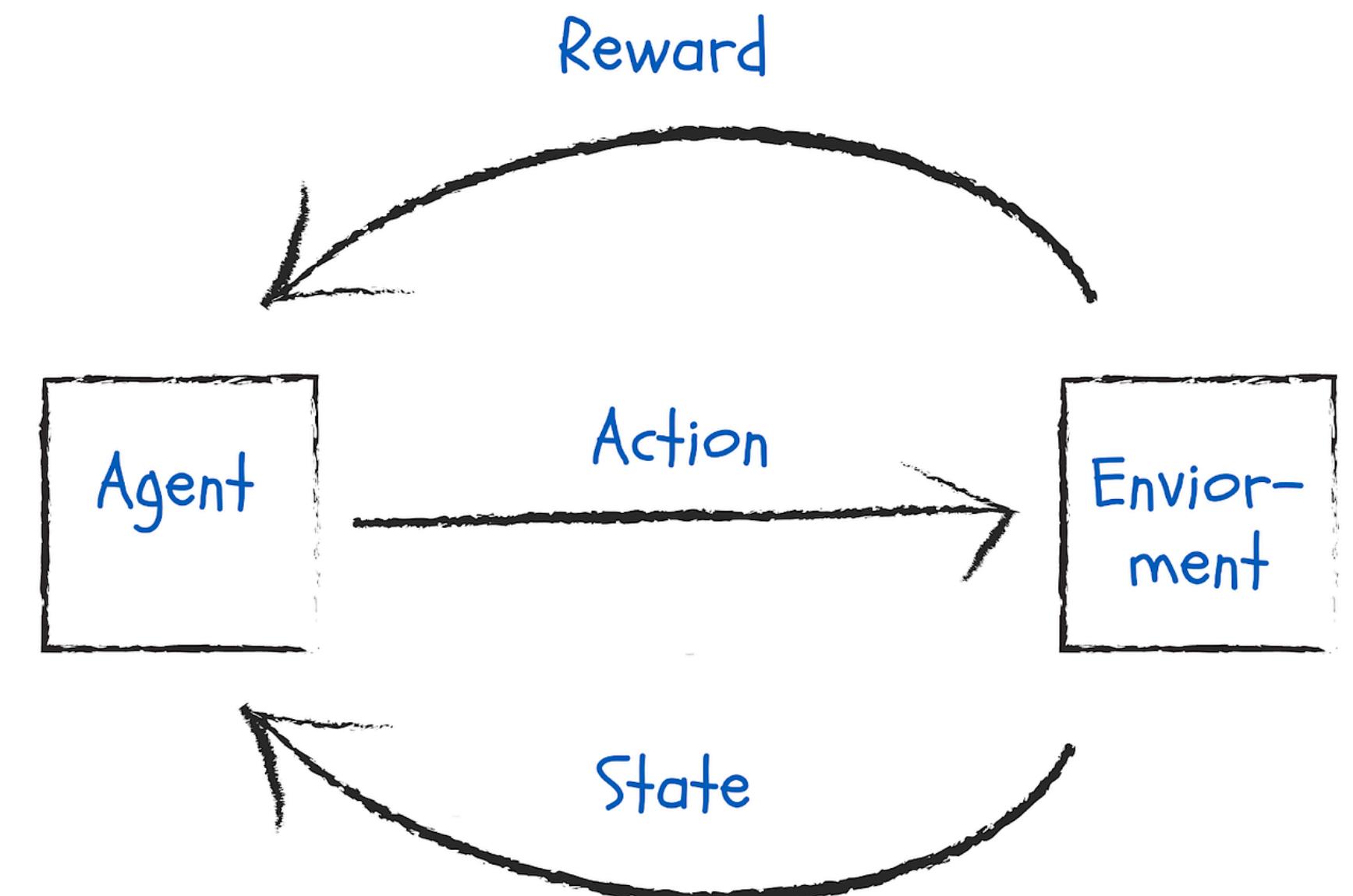
Agents Properties

- **Weak notion of agency**
 - Autonomy
 - Reactivity
 - Pro-activeness
 - Social ability
- **Stronger notion of agency**
 - mobility
 - veracity
 - benevolence
 - rationality

Wooldridge, M. J., N. R. Jennings. Intelligent agents: theory and practice. *Knowl. Eng. Rev.*, 10(2):115–152, 1995

Reinforcement Learning Agents

- learn to make decisions by interacting with an environment, aiming to maximize a cumulative reward signal
- Examples:
 - AlphaGO
 - Autonomous vehicles



LLM-based agents

Non-agentic workflow (zero-shot):

Please type out an essay on topic X from start to finish in one go, without using backspace.



Agentic workflow:

Write an essay outline on topic X

Do you need any web research?

Write a first draft.

Consider what parts need revision or more research.

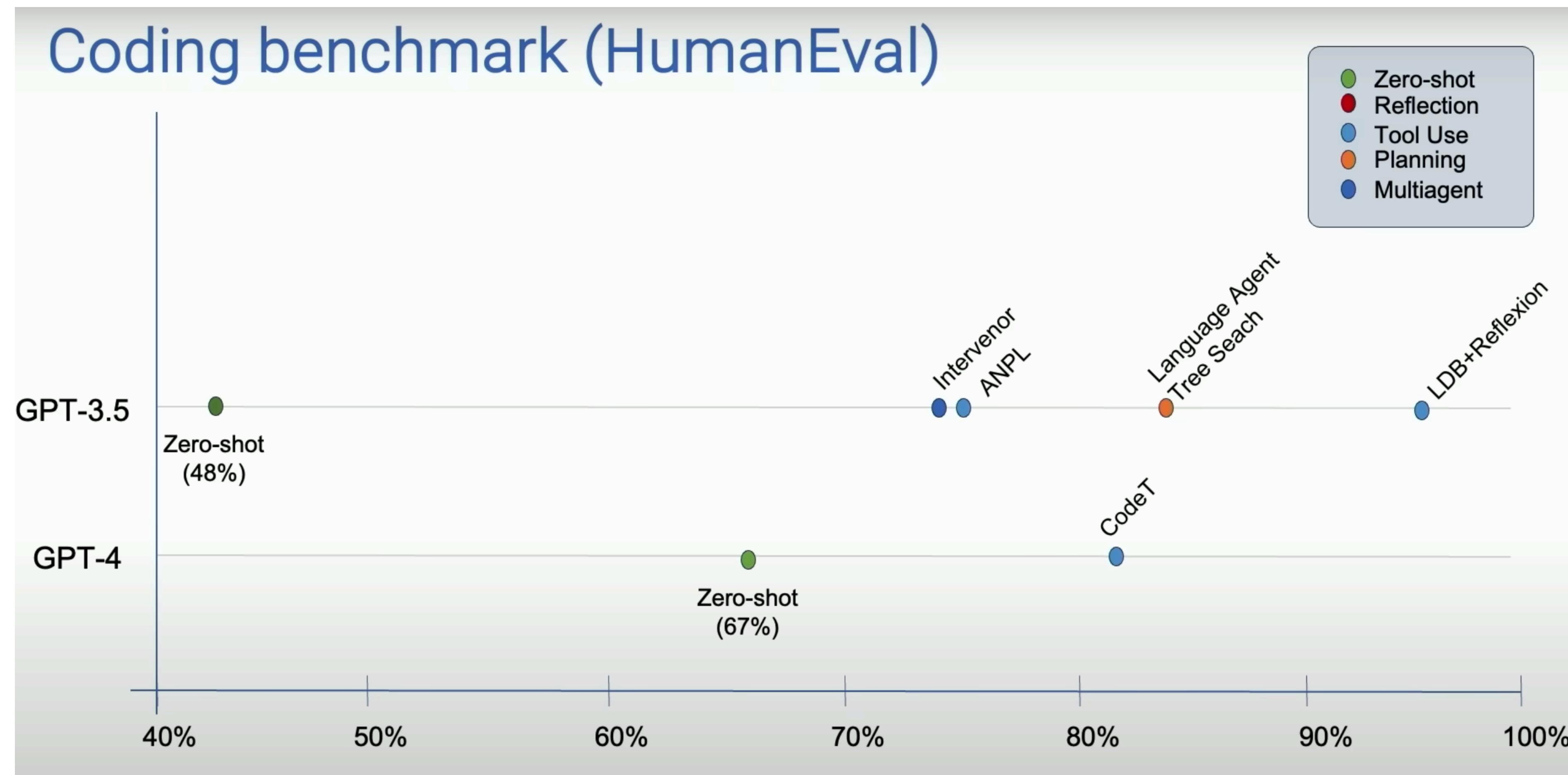
Revise your draft.

....



Andrw Ng talk: agentic design pattern on Youtube, <https://www.youtube.com/watch?v=sal78ACtGTc>

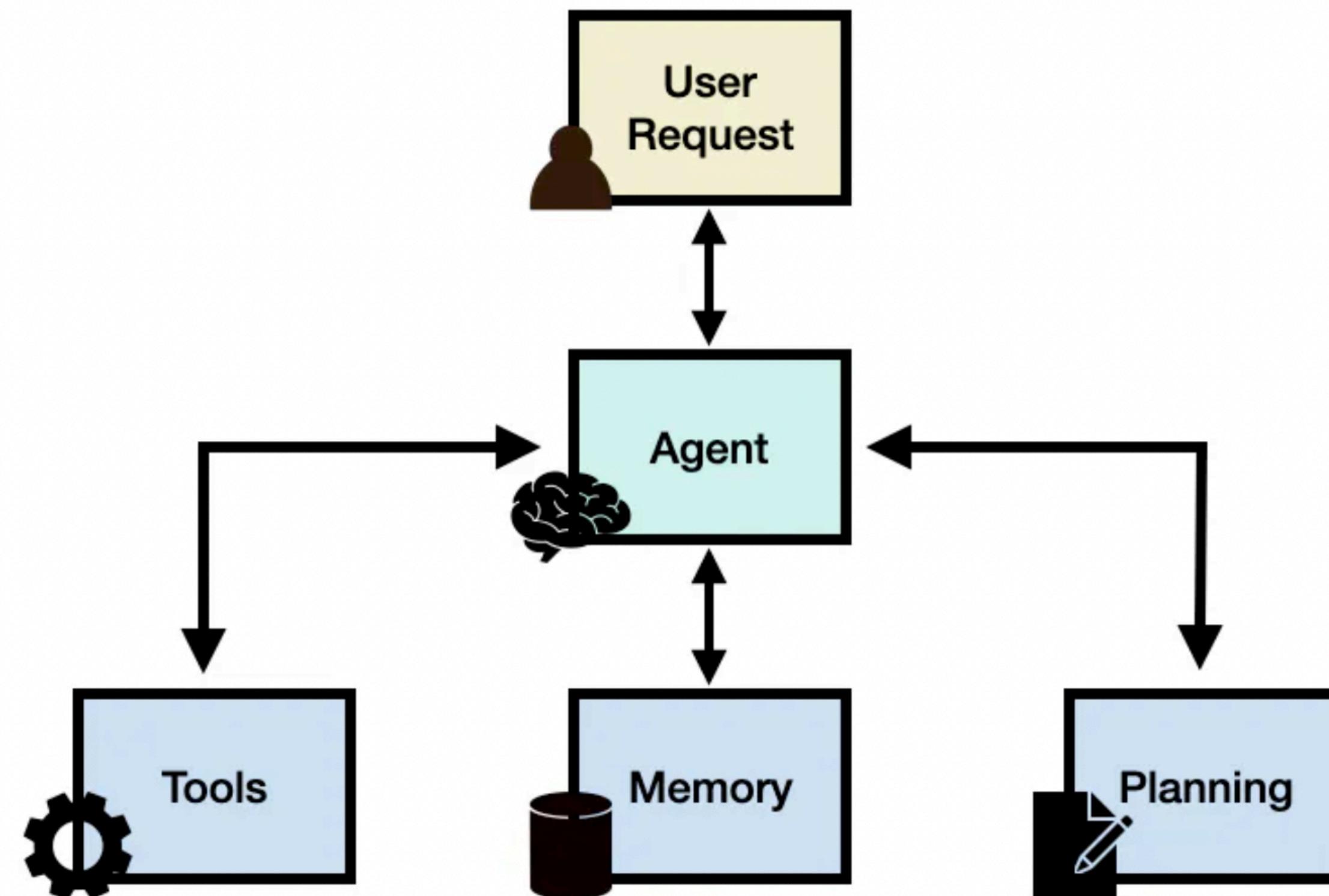
LLM-based agents



Andrw Ng talk: agentic design pattern on Youtube, <https://www.youtube.com/watch?v=sal78ACtGTc>

LLM-based agents

architecture design



LLM-based agents

Actions - Tool Use



LLM-based agents

Toolformer

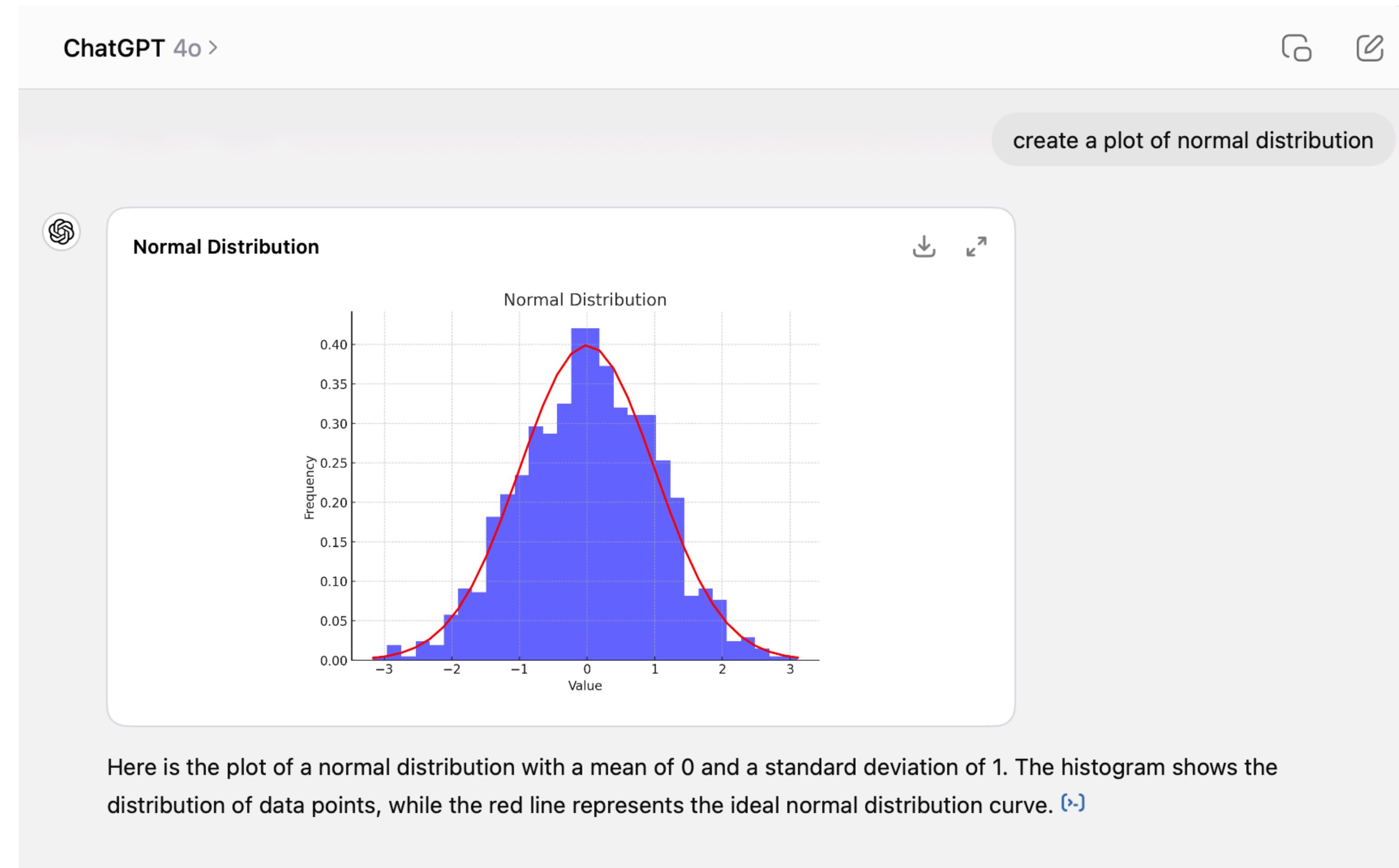
- fine-tune a LM to learn to use external tool APIs
- Toolformers APIs:
 - Calendar API
 - Calculator
 - Wikipedia Search
 - Question Answering system

What day of the week is it today?

I am sorry, but as a language model AI, I do not have the capability to access current time or date information. Please check your device or a reliable source [...]

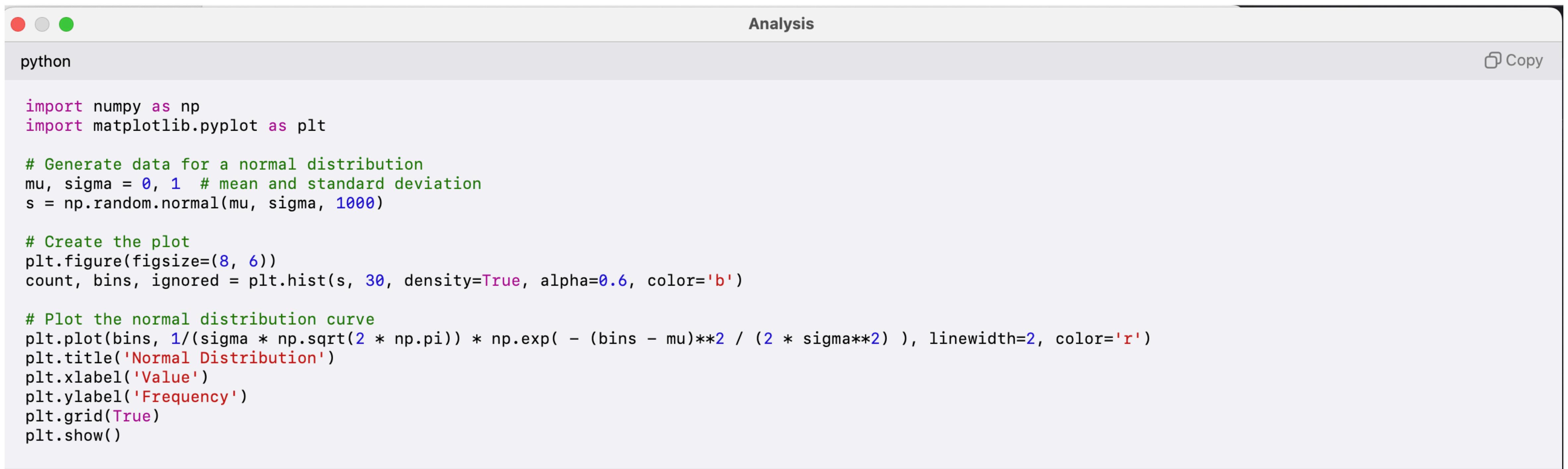
LLM-based agents

Actions - Tool Use



LLM-based agents

Actions - Tool Use



The screenshot shows a Jupyter Notebook cell titled "Analysis". The code in the cell generates a histogram of a normal distribution with a superimposed normal distribution curve.

```
python

import numpy as np
import matplotlib.pyplot as plt

# Generate data for a normal distribution
mu, sigma = 0, 1 # mean and standard deviation
s = np.random.normal(mu, sigma, 1000)

# Create the plot
plt.figure(figsize=(8, 6))
count, bins, ignored = plt.hist(s, 30, density=True, alpha=0.6, color='b')

# Plot the normal distribution curve
plt.plot(bins, 1/(sigma * np.sqrt(2 * np.pi)) * np.exp( - (bins - mu)**2 / (2 * sigma**2) ), linewidth=2, color='r')
plt.title('Normal Distribution')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```

LLM-based agents

Actions - Tool Use

The screenshot shows the LangChain website interface. The top navigation bar includes the LangChain logo, a search bar, and links for Components, Integrations, Guides, API Reference, More, version v0.1, and user profile. The left sidebar has sections for Providers (Anthropic, AWS, Google, Hugging Face, Microsoft, OpenAI, More) and Components (Chat models, LLMs, Embedding models, Document loaders, Document transformers, Vector stores, Retrievers, Tools). The 'Tools' section is currently selected. The main content area displays six tool cards: Alpha Vantage, Apify, ArXiv, AWS Lambda, Shell (bash), and Bearly Code Interpreter.

- Providers**
 - Anthropic
 - AWS
 - Google
 - Hugging Face
 - Microsoft
 - OpenAI
 - More >
- Components**
 - Chat models >
 - LLMs >
 - Embedding models >
 - Document loaders >
 - Document transformers >
 - Vector stores >
 - Retrievers >
 - Tools** > (highlighted)

Tools

- Alpha Vantage**
Alpha Vantage Alpha Vantage provides realtime and histo...
- Apify**
This notebook shows how to use the Apify integration for ...
- ArXiv**
This notebook goes over how to use the arxiv tool with an ...
- AWS Lambda**
Amazon AWS Lambda is a serverless computing service p...
- Shell (bash)**
Giving agents access to the shell is powerful (though risk...)
- Bearly Code Interpreter**
Bearly Code Interpreter allows for remote execution of co...

LLM-based agents

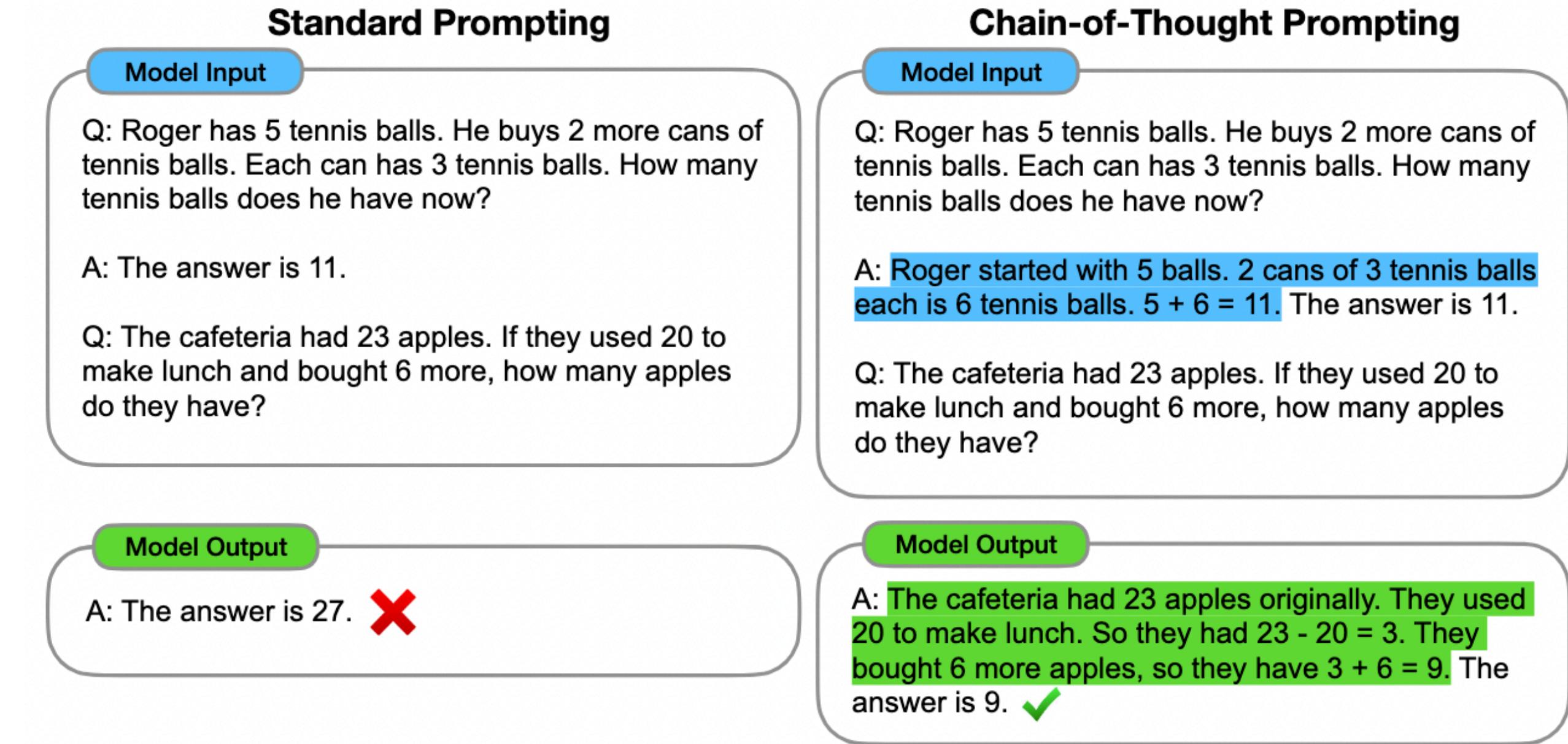
Planning

- Chain of Thoughts
- ReAct
- self reflection

LLM-based agents

Planning - CoT

- decompose hard tasks
- Think step by step

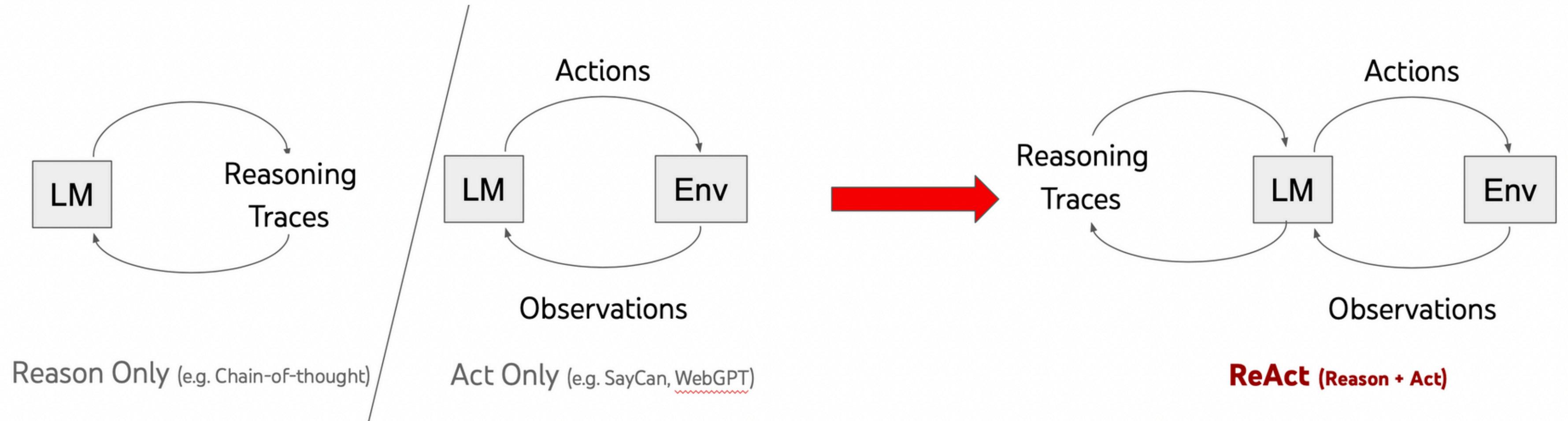


"As for limitations, we first qualify that although chain of thought emulates the thought processes of human reasoners, this does not answer whether the neural network is actually "reasoning," which we leave as an open question"

LLM-based agents

Planning - self reflection - ReAct

- allows autonomous agents to improve iteratively by refining past action decisions and correcting previous mistakes



Yao, Shunyu, et al. "React: Synergizing reasoning and acting in language models." *arXiv preprint arXiv:2210.03629* (2022).

LLM-based agents

Planning - self reflection - ReAct

ReAct

Claim: Reign Over Me is an American film made in 2010.

Thought 1: I need to search for Reign Over Me and find if it is an American film made in 2010.

Action 1: Search[Reign Over Me]

Observation 1: Reign Over Me ...

Thought 2: The observation says that it is an American film made in 2007, so it is not made in 2010.

Action 2: Finish[REFUTES]

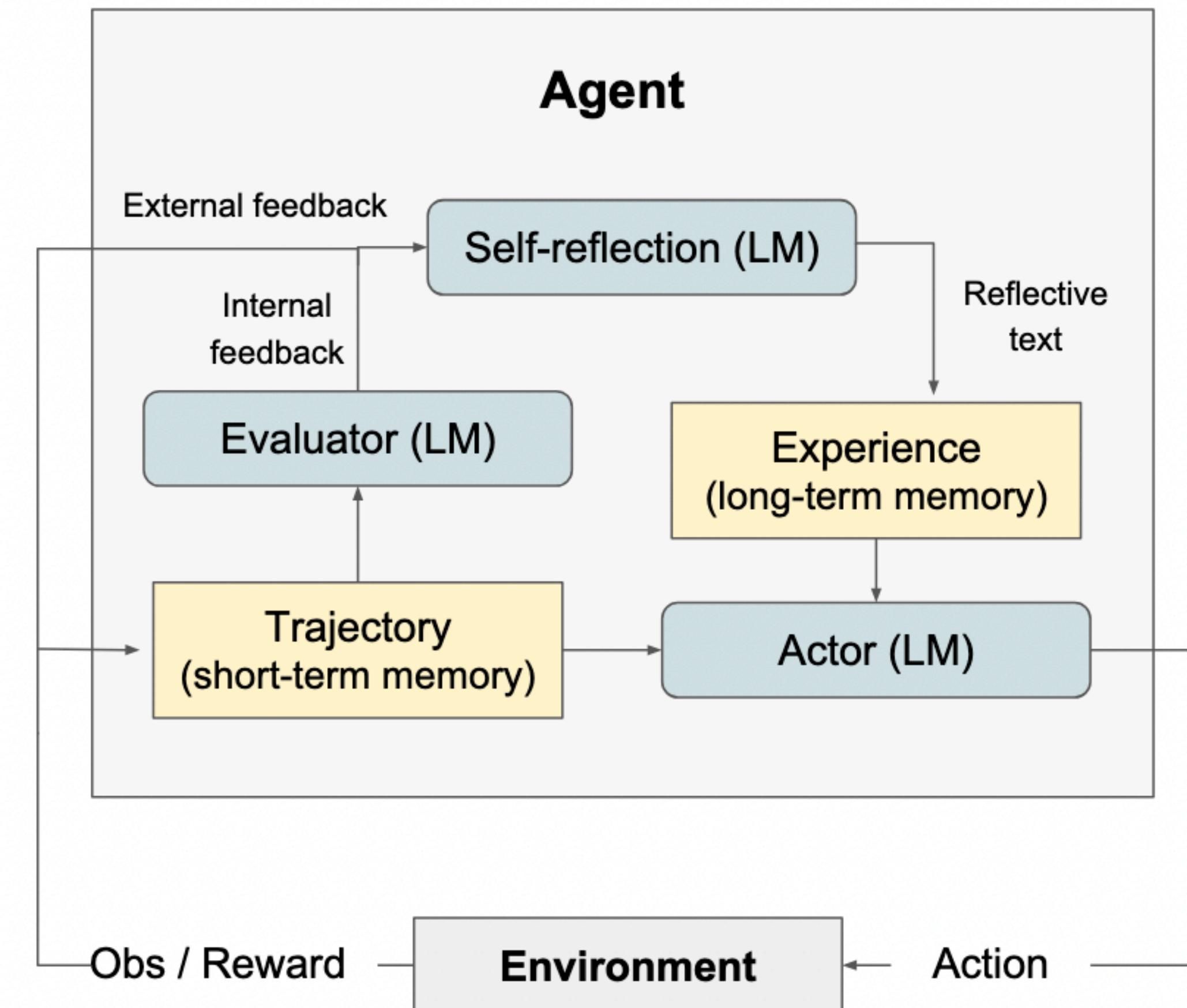
Observation 2: Episode finished

Yao, Shunyu, et al. "React: Synergizing reasoning and acting in language models." *arXiv preprint arXiv:2210.03629* (2022).

LLM-based agents

Planning - self reflection - Reflexion

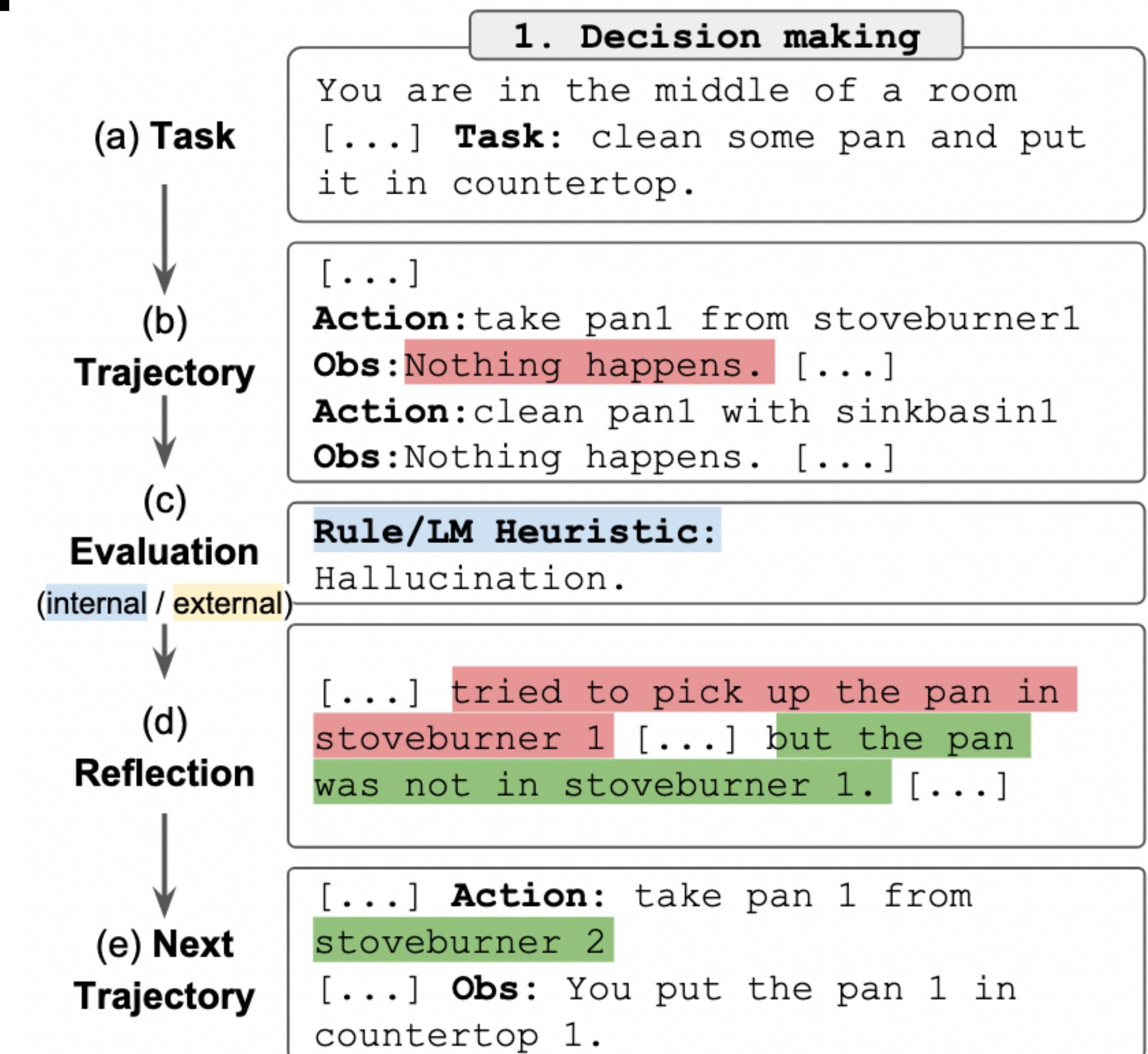
- Reinforce language agents not by updating weights but instead through linguistic feedback



LLM-based agents

Planning - self reflection - Reflexion

- **Actor:** CoT, ReAct
- **Evaluator:** LLM eval, Binary, pre-defined heuristics
- **Self-Reflection:** converts binary or scalar feedback into verbal feedback in the form of a textual summary



LLM-based agents

Planning - self reflection - Reflexion

Trial #1: Question: What profession does John Lanchester and Alan Dean Foster have in common?

Thought 1: Let's think step by step. John Lanchester is a novelist, journalist, and critic. Alan Dean Foster is a novelist and screenwriter. So the profession John Lanchester and Alan Dean Foster have in common is novelist and screenwriter.

Action 1: Finish[novelist and screenwriter]

Observation 1: Answer is INCORRECT

Reflection: My reasoning for the profession John Lanchester and Alan Dean Foster have in common failed because I incorrectly assumed that they both had the same profession. In the future, when attempting this question, I should focus on better researching the individual backgrounds of both authors to ensure that I am accurately identifying their professions. Additionally, I should consider the possibility that they may have multiple professions in common.

Trial #2: Question: What profession does John Lanchester and Alan Dean Foster have in common?

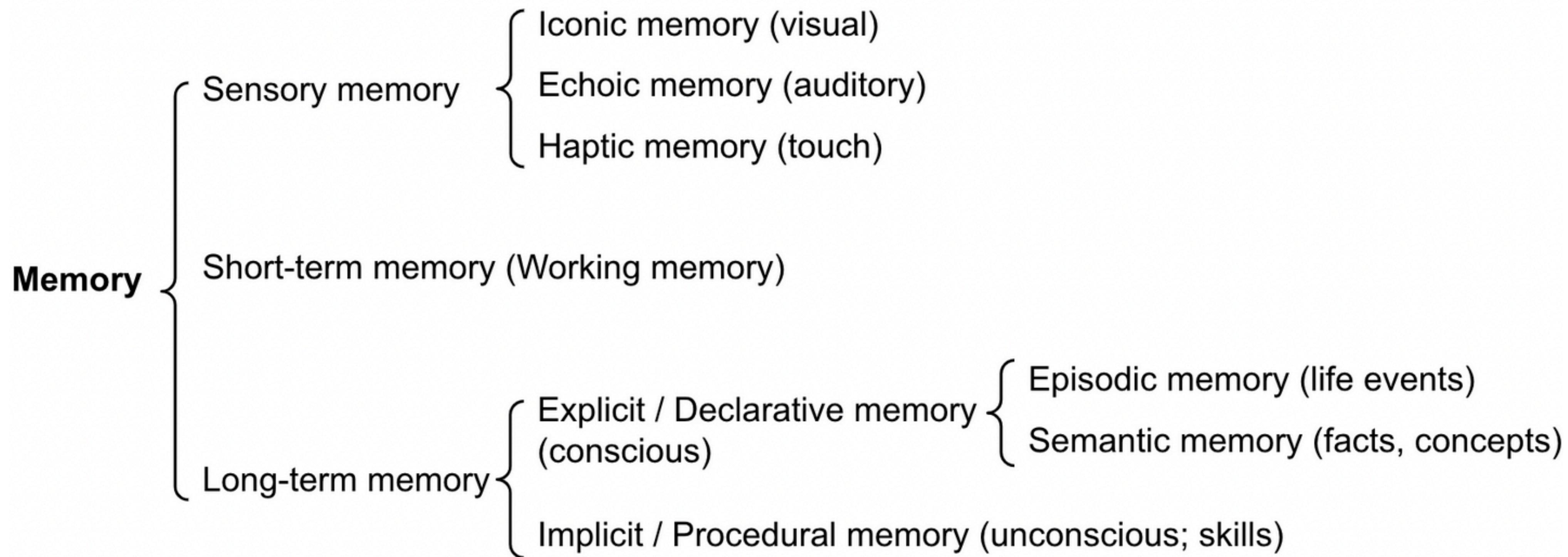
Thought 1: Let's think step by step. John Lanchester is a novelist, journalist, and financial journalist. Alan Dean Foster is a novelist and screenwriter. So the profession John Lanchester and Alan Dean Foster have in common is novelist.

Action 1: Finish[novelist]

Observation 1: Answer is CORRECT

LLM-based agents

Memory



LLM-based agents

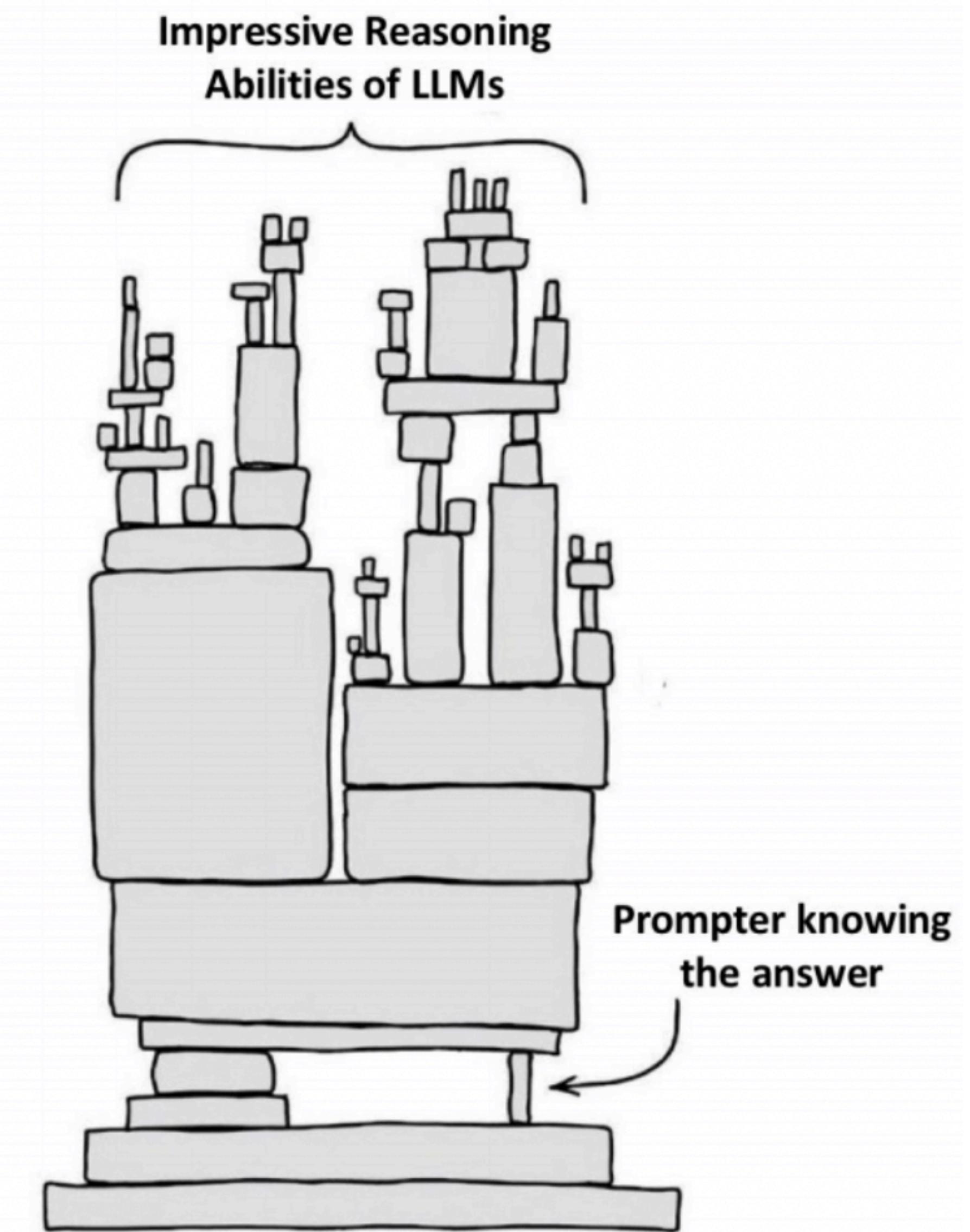
Memory

- Short-term memory:
 - In-context learning. It is short and finite, as it is restricted by the finite context window length of Transformer
- Long-term memory:
 - the external vector store that the agent can attend to at query time, accessible via fast retrieval.

LLM-based agents

Challenges

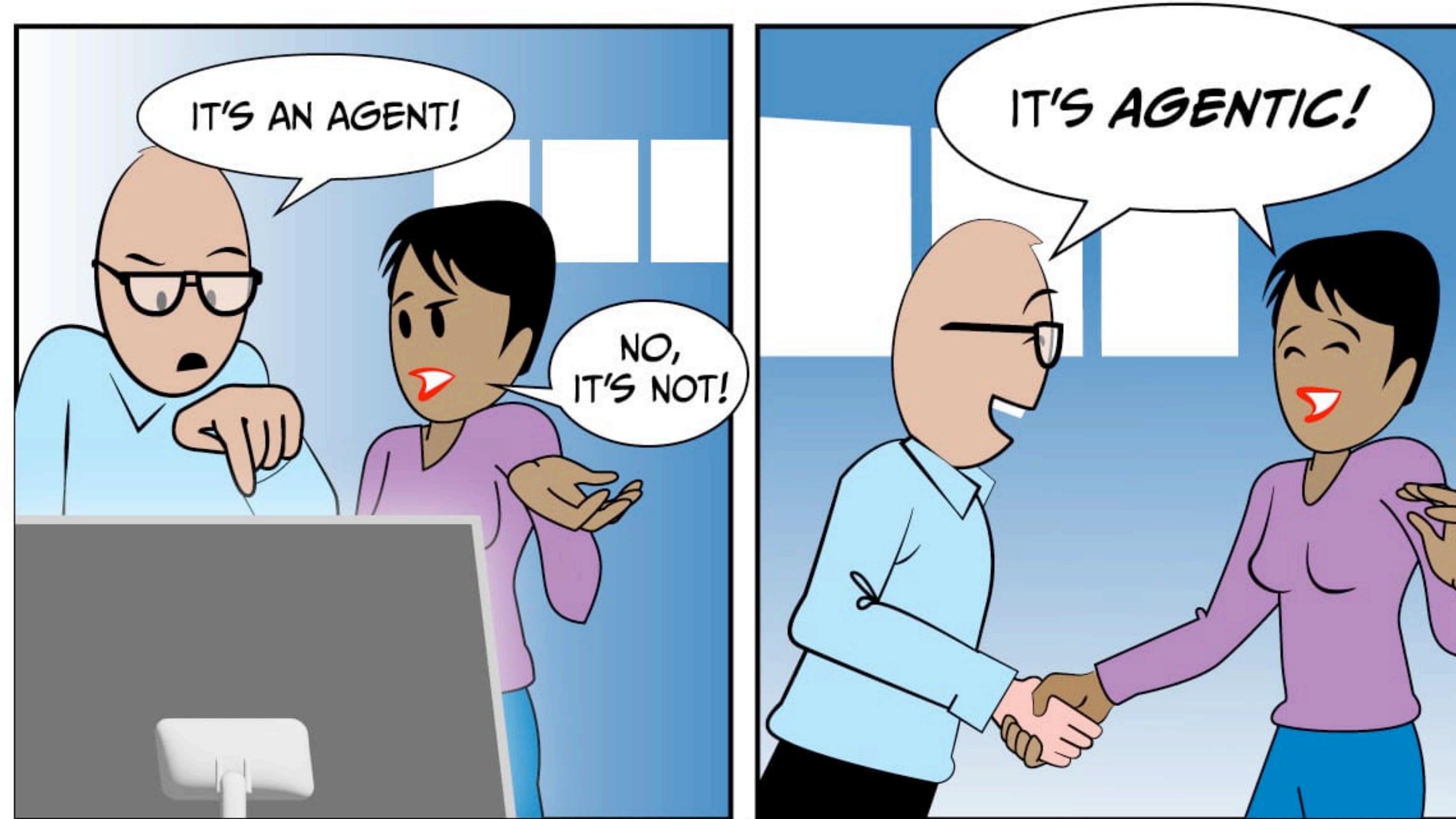
- Finite context length
- Challenges in long-term planning and task decomposition
- Reliability of natural language interface



Kambhampati, Subbarao. "Can large language models reason and plan?." *Annals of the New York Academy of Sciences* 1534.1 (2024): 15-18.

LLM-based agents

Are they truly agents?



<https://www.deeplearning.ai/the-batch/welcoming-diverse-approaches-keeps-machine-learning-strong/>

Thank You

QA time