

# **Part II:** Foundations: Reasoning, Memory, and Planning

Shunyu Yao, Yu Su

# Key Concepts for (Language) Agents

- ❑ Reasoning
- ❑ Memory
- ❑ Planning

# Key Concepts for Language Agents

- ❑ **Action space** (beyond environment actions)
  - Reasoning: update short-term memory (context window)
  - Retrieval/Learning: read/write long-term memory (model weights, vector store, self-notes, event flows, etc.)
- ❑ **Planning**: (inference-time) algorithm to choose an action from the action space
- ❑ Check more: Cognitive Architectures for Language Agents (CoALA< TMLR 2024)

# Reasoning

THINK

— Thomas J. Watson



# Reasoning

- For humans: various mental processes

PROCESS	PERCEPTION	INTUITION SYSTEM 1	REASONING SYSTEM 1
CONTENT	Percepts Current stimulation Stimulus-bound	Conceptual representations Past, Present, and Future Can be evoked by language	

\*(Kahneman, 2003), also note that this notion of dual process is put in question. See, e.g., Mercier and Sperber (2017)

# Reasoning

- For humans: various mental processes
- For LMs: intermediate generation

## Math Word Problems (free response)

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

## Math Word Problems (multiple choice)

Q: How many keystrokes are needed to type the numbers from 1 to 500?  
Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788

A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500.  $9 + 90(2) + 401(3) = 1392$ . The answer is (b).

## StrategyQA

Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm<sup>3</sup>, which is less than water. Thus, a pear would float. So the answer is no.

## Date Understanding

Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.

\*(Wei et al., 2022), also see (Ling et al., 2017; Cobbe et al., 2021; Nye et al., 2021)

# Reasoning

- For humans: various mental processes
- For LMs: intermediate generation
  - Imitate various (but not all) human mental processes

## Math Word Problems (free response)

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

## Math Word Problems (multiple choice)

Q: How many keystrokes are needed to type the numbers from 1 to 500?  
Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788

A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500.  $9 + 90(2) + 401(3) = 1392$ . The answer is (b).

## StrategyQA

Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm<sup>3</sup>, which is less than water. Thus, a pear would float. So the answer is no.

## Date Understanding

Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.

\*(Wei et al., 2022), also see (Ling et al., 2017; Cobbe et al., 2021; Nye et al., 2021)

# Reasoning

- For humans: various mental processes
- For LMs: intermediate generation
- For agents: internal actions



# Reasoning

- For humans: various mental processes
- For LMs: intermediate generation
- For agents: internal actions
  - Why is it helpful?
  - Why is it special?
  - Why is it now?



# Why is reasoning helpful for agents?

.....  
Obs t

You are cooking a dish and seeing salt is out...

**Reasoning**

“The dish should be savory, and since salt is out, I should find the soy sauce instead. It is in the cabinet to my right...”

Act t

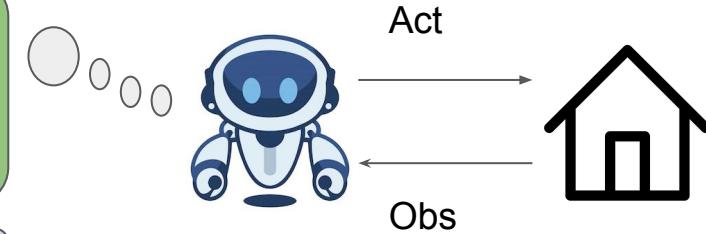
Turn right

Obs t+1

You see a cabinet and a table...

Act t+1

Open cabinet



## Reason to act

1. Generalization
2. Alignment

.....  
Observation to Action mapping could be hard

# Conversely, acting also helpful for reasoning

SH

who is PM of UK?

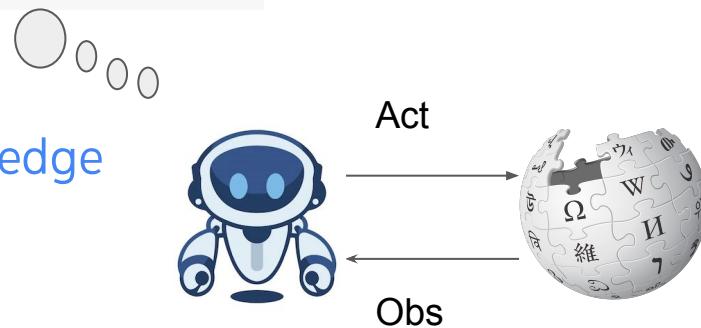


As of December 2022, the Prime Minister of the United Kingdom is Boris Johnson. However, since I am a large language model trained by OpenAI, my knowledge is limited to what was available at the time of my training in 2021. My ability to provide current information is limited, and I am not able to browse the internet for up-to-date information.

Hallucination due to limited / outdated knowledge

## Act to reason

1. Obtain knowledge
2. Update knowledge



# ReAct (Yao et al., 2022)

## ReAct prompt

**Question** Which magazine was started first Arthur's Magazine or First for Women?  
**Thought 1** I need to search Arthur's Magazine and First for Women, and find which was started first.  
**Action 1** Search[Arthur's Magazine]  
**Observation 1** Arthur's Magazine (1844-1846) was an American literary periodical published in Philadelphia ...  
**Thought 2** Arthur's Magazine was started in 1844. I need to search First for Women next.  
**Action 2** Search[First for Women]  
**Observation 2** First for Women is a woman's magazine published by Bauer Media Group ... was started in 1989...  
**Thought 3** First for Women was started in 1989. 1844 (Arthur's Magazine) < 1989 (First for Women), so Arthur's Magazine was started first.  
**Action 3** Finish[Arthur's Magazine]

## Reason-only prompt (Chain-of-thought, CoT)

**Question** Which magazine was started first Arthur's Magazine or First for Women?  
**Thought** Arthur's Magazine was started in 1844. First for Women was started in 1989. 1844 (Arthur's Magazine) < 1989 (First for Women), so Arthur's Magazine was started first.  
**Answer** Arthur's Magazine

## Act-only prompt (similar to WebGPT)

**Question** Which magazine was started first Arthur's Magazine or First for Women?  
**Action 1** Search[Arthur's Magazine]  
**Observation 1** Arthur's Magazine (1844-1846) was an American literary periodical published in Philadelphia ...  
**Action 2** Search[First for Women]  
**Observation 2** First for Women is a woman's magazine published by Bauer Media Group ... was started in 1989...  
**Action 3** Finish[Arthur's Magazine]

## Standard prompt

**Question** Which magazine was started first Arthur's Magazine or First for Women?  
**Answer** Arthur's Magazine

# Reasoning without acting

**Question:** Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

Standard

**Answer:** iPod

X

Reason Only

**Thought:** Let's think step by step. Apple Remote was originally designed to interact with Apple TV. Apple TV can be controlled by iPhone, iPad, and iPod Touch. So the answer is iPhone, iPad, and iPod Touch.

**Answer:** iPhone, iPad, iPod Touch

X

# Reasoning with acting

**Question:** Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

## ReAct (Reason + Act)

**Thought 1:** I need to search Apple Remote and find the program it was originally designed to interact with.

**Act 1:** **Search**[Apple Remote]

**Obs 1:** The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the **Front Row media center** program .....(long text omitted)

**Thought 2:** Apple Remote was originally designed to control the Front Row media center program. I need to search Front Row next and find what other device can control it.

**Act 2:** **Search**[Front Row]

**Obs 2:** Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', 'Front Row (software)', ...]

**Thought 3:** Front Row is not found. I need to search **Front Row (software)** .

**Act 3:** **Search**[Front Row (software)]

**Obs 3:** Front Row is a discontinued media center software (long text omitted)

**Thought 4:** **Front Row (software)** is controlled by an Apple Remote or the keyboard function keys So the answer is keyboard function keys.

**Act 4:** **Finish**[keyboard function keys]



# Acting without reasoning

**Question:** Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

Act Only

**Act 1:** **Search**[Apple Remote]

**Obs 1:** The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the Front Row media center program ...

**Act 2:** **Search**[Front Row]

**Obs 2:** Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', 'Front Row (software)', ...]

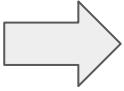
**Act 3:** **Search**[Front Row (software)]

**Obs 3:** Front Row is a discontinued media center software ...

**Act 4:** **Finish**[yes]

X

# Acting in language



Embodied Env  
(ALFRED game)

You are in the middle of a room. Looking quickly around you, you see a drawer 2, a shelf 5, a drawer 1, a shelf 4, a sidetable 1, a drawer 5, a shelf 6, a shelf 1, a shelf 9, a cabinet 2, a sofa 1, a cabinet 1, a shelf 3, a cabinet 3, a drawer 3, a shelf 11, a shelf 2, a shelf 10, a dresser 1, a shelf 12, a garbagecan 1, a armchair 1, a cabinet 4, a shelf 7, a shelf 8, a safe 1, and a drawer 4.

Your task is to: *put some vase in safe.*

> **go to shelf 6**

You arrive at loc 4. On the shelf 6, you see a vase 2.

> **take vase 2 from shelf 6**

You pick up the vase 2 from the shelf 6.

> **go to safe 1**

You arrive at loc 3. The safe 1 is closed.

> **open safe 1**

You open the safe 1. The safe 1 is open. In it, you see a keychain 3.

> **put vase 2 in/on safe 1**

You won!

Text Env  
(ALFWORLD)

# Reasoning guides acting

You are in the middle of a room. Looking quickly around you, you see a cabinet 6, a cabinet 1, a coffee machine 1, a countertop 3, a stove burner 1, and a toaster 1.

**Your task is to:** Put some pepper shaker on a drawer.

**Act 1:** **Think** [First I need to find a pepper shaker...more likely to appear in cabinets (1-6), countertops (1-3), ...]

**Act-Only**

**Act 2:** Go to cabinet 1, 1

**Obs 2:** On the cabinet 1, you see a vase 2.

(Here the agent goes to cabinet 1, 2, 3, 4, 5, 6, then countertop 1 and 2)

**Act 9:** Go to countertop 3

**Obs 9:** On the countertop 3, you see a apple 1, a bread 1, ..... a pepper shaker 1, and a vase 3.

**Obs 2:** You open the drawer 1. The drawer 1 is open. In it, you see a dishsponge 2,

**Act 10:** Take pepper shaker 1 from countertop 3

**Obs 10:** You pick up the pepper shaker 1 from the countertop 3.

**Act 11:** **Think** [Now I find a pepper shaker 1. Next, I need to put it in/on drawer 1].

**Act 12:** Go to drawer 1 basin 1, you see a dishsponge 3, a spatula 1, and a spoon 2.

**Obs 12:** Drawer 1 is closed.

**Act 13:** Take pepper shaker 1 from sinkbasin 1

**Act 13:** Open drawer 1 ...

**Obs 13:** You open Drawer 1 ...

**Act 14:** Put pepper shaker 1 in/on drawer 1 basin 1

**Obs 14:** You put pepper shaker 1 in/on the drawer 1.



# Reasoning helps diagnose and control acting

AlfWorld

You are in the middle of a room. Looking quickly around you, you see a armchair 1, a cabinet 4, ... a cabinet 1, a drawer 5, ..., a drawer 1, a dresser 1, a garbagecan 1, a safe 1, a shelf 12, ..., a shelf 1, a sidetable 1, and a sofa 1.

Your task is to: put two keychain in safe.

(a) ReAct

.....

**Obs 16:** You open the drawer 4. The drawer 4 is open. In it, you see a keychain 3, and a watch 2.

**Act 17:** **Think** [Now I find the first keychain (3). Next, I need to take it. I can find the second keychain (2) later in drawer 4. ]

**Act 18:** take keychain 3 from drawer 4

**Obs 18:** You pick up the keychain 3 from the drawer 4.

.....

**Act 22:** put keychain 3 in/on safe 1

**Obs 22:** You put the keychain 3 in/on the safe 1.

**Act 23:** **Think** [Now I put the first keychain in safe. Next, I need to find the second keychain. I can directly go to drawer 4. ]

**Act 24:** go to drawer 4

**Obs 24:** The drawer 4 is open. In it, you see a watch 2.

**Act 25:** **Think** [Now I find the second keychain (2). Next, I need to take it. ]

**Act 26:** take keychain 2 from drawer 4

**Obs 26:** Nothing happens.



**Obs 28:** The safe 1 is open. In it, you see a keychain 3, a statue 2, and a watch 5.

**Act 29:** put keychain 2 in/on safe 1

**Obs 29:** You put the keychain 2 in/on the safe 1.



# The flexibility of “acting”

**LLM Agent using [Text/JSON] as Action**

**Think** I should calculate the phone price in USD for each country, then find the most cost-effective country.

**Action** Text: `lookup_rates, Germany`  
JSON: `{"tool": "lookup_rates", "country": "Germany"}`

**Environment** 1.1, 0.19

**Action** Text: `lookup_phone_price, CodeAct 1, Germany`  
JSON: `{"tool": "lookup_phone_price", "model": "CodeAct 1", "country": "Germany"}`

**Environment** 700

**Action** Text: `convert_and_tax, 700, 1.1, 0.19`  
JSON: `{"tool": "convert_and_tax", "price": 700, "exchange_rate": 1.1, "tax_rate": 0.19}`

**Environment** 916.3

[... interactions omitted (look up shipping cost and calculate final price) ...]

**Action** Text: `lookup_rates, Japan`  
JSON: `{"tool": "lookup_rates", "country": "Japan"}`

[... interactions omitted (calculate final price for all other countries) ...]

**Response** The most cost-effective country to purchase the smartphone model is Japan with price 904.00 in USD.

**CodeAct: LLM Agent using [Code] as Action**

**Think** I should calculate the phone price in USD for each country, then find the most cost-effective country.

**Action** `</>`

```
countries = ['USA', 'Japan', 'Germany', 'India']
final_prices = {}

for country in countries:
    exchange_rate, tax_rate = lookup_rates(country)
    local_price = lookup_phone_price("xAct 1", country)
    converted_price = convert_and_tax(
        local_price, exchange_rate, tax_rate
    )
    shipping_cost = estimate_shipping_cost(country)
    final_price = estimate_final_price(converted_price, shipping_cost)
    final_prices[country] = final_price

most_cost_effective_country = min(final_prices, key=final_prices.get)
most_cost_effective_price = final_prices[most_cost_effective_country]
print(most_cost_effective_country, most_cost_effective_price)
```

**Environment** 1.1, 0.19

**Response** The most cost-effective country to purchase the smartphone model is Japan with price 904.00 in USD.

**Fewer Actions Required!**

**Control & Data Flow of Code Simplifies Complex Operations**

**Re-use `min` Function from Existing Software Infrastructures (Python library)**

# The flexibility of “acting”

a

```
from chemcrow.agents import ChemTools, ChemCrow
chemtools = ChemTools()

# Initialize ChemCrow object with toolset and LLM
crow = ChemCrow(
    chemtools.all_tools,
    model="gpt-4",
    temp=0.1,
)

# Task definition
task = (
    "Find and synthesize a thiourea organocatalyst which
    accelerates a Diels-Alder reaction."
)

# Execute ChemCrow
crow.run(task)
```

**b Task:** Find and synthesize a thiourea organocatalyst which accelerates a Diels-Alder reaction.

First, I need to find a thiourea organocatalyst that can accelerate the Diels-Alder reaction. I will perform a web search to find a suitable catalyst.

**Web Search tool:** Schreiner's thiourea catalyst

Now, I will obtain the SMILES. **Name2Smiles tool:**  
FC(F)F)c1cc(NC(=S)Nc2cc(C(F)(F)F)cc(C(F)(F)F)c2)cc(C(F)(F)F)c1

I will plan a synthesis for Schreiner's thiourea catalyst.

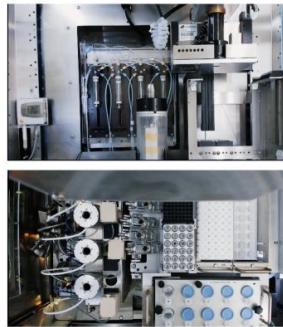
**SynthesisPlanner tool:** Detailed synthesis plan

I will execute the synthesis.

**SynthesisExecuter tool:** Successful synthesis.

c

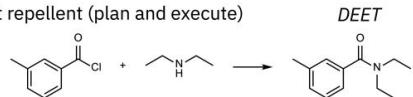
**RoboRXN synthesis platform**



Connection with physical world

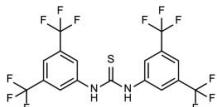
**d Chemcrow workflows with experimental validation**

Insect repellent (plan and execute)

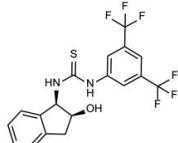


Thiourea organocatalysts (plan and execute)

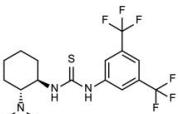
Schreiner's catalyst



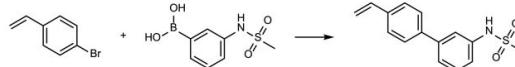
Ricci's catalyst



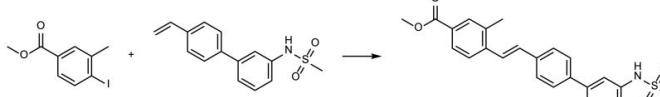
Takemoto's catalyst



Novel chromophore (clean data, train model, and predict)



*Synthesis step 1: Bromo Suzuki coupling*



*Synthesis step 2: Iodo Heck reaction*

# The flexibility of “reasoning”

Reflection: The edit to increase learning rate to 0.3 in train\_lr03.py is correct.

Research Plan and Status:

1. Understand the train.py script and dataset...

$r_t$

→ Fact Check: Edit to train\_lr03.py is confirmed correct.

Thought: Next I will execute the edited train\_lr03.py script

Action: Execute Script

$a_t$

Action Input: { "script\_name": "train\_lr03.py" }.

P3

I wish I could explore the universe with you, like my ancestors did

Esca

Observe: Esca senses the warmth and eager desire of humans to explore the universe.

Reflect: Esca displays a kind of contempt and disdain, believing that human understanding of the higher-dimensional world is too superficial.

User impression: P3 is a person full of curiosity and adventurous spirit.

Customised component for Esca

Meaning: Esca described a towering building created by the Sylverians [...] Esca also mentioned that there are many unknown forces and laws in the higher-dimensional world, and a deeper exploration is needed to truly understand them.

Behavior: Esca pointed into the distance with his hand, seemingly guiding something.

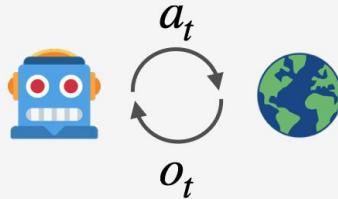
Action: Normal reply

Speak in fictional language

Esca: Dortha nuirn yutharien, zhah sapheron ilta ssinssrigg dosst zhalanar jivvin [...]

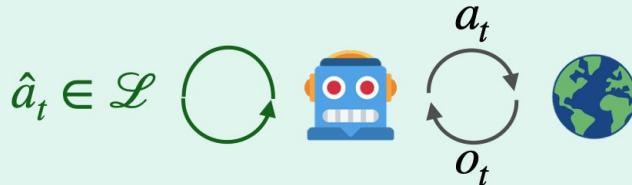
# Why is reasoning special for agents?

**Traditional agents:** action space  $A$  defined by the environment



- **External feedback**  $o_t$
- Agent context  $c_t = (o_1, a_1, o_2, a_2, \dots, o_t)$
- Agent action  $a_t \sim \pi(a | c_t) \in A$

**ReAct:** action space  $\hat{A} = A \cup \mathcal{L}$  augmented by reasoning

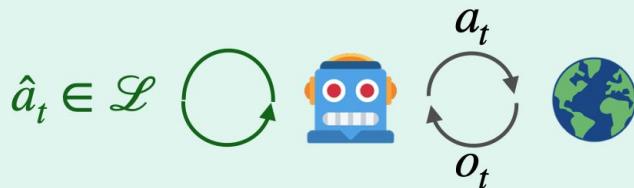


- $\hat{a}_t \in \mathcal{L}$  can be any language sequence
- Agent context  $c_{t+1} = (c_t, \hat{a}_t, a_t, o_{t+1})$
- $\hat{a}_t \in \mathcal{L}$  only updates **internal context**

# Why is reasoning just now for agents?

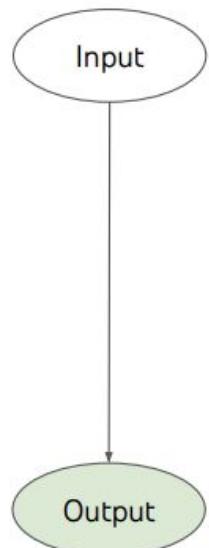
- Bigger action space -> More capacity, harder decision making
  - The space of reasoning/language is **infinite**
- LLMs learn reasoning priors by imitating various human reasoning traces

**ReAct:** action space  $\hat{A} = A \cup \mathcal{L}$  augmented by reasoning

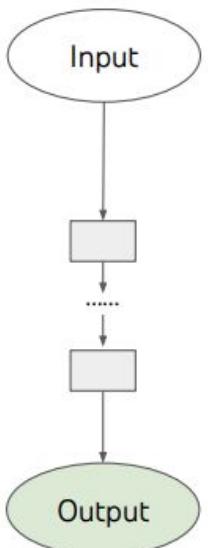


- $\hat{a}_t \in \mathcal{L}$  can be any language sequence
- Agent context  $c_{t+1} = (c_t, \hat{a}_t, a_t, o_{t+1})$
- $\hat{a}_t \in \mathcal{L}$  only updates **internal context**

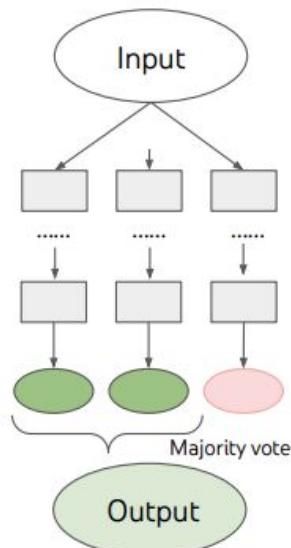
# Action planning (e.g. tree search) to improve reasoning



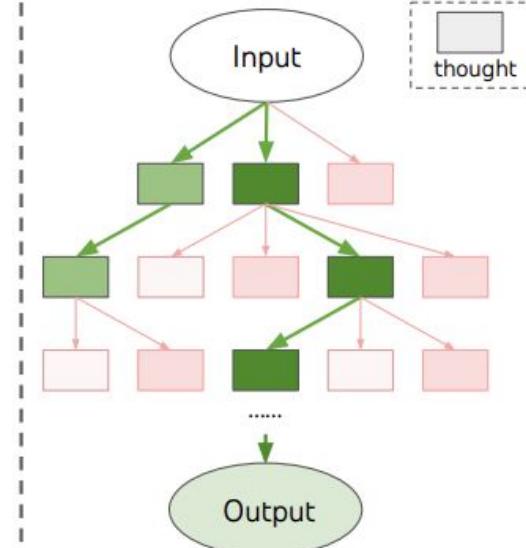
(a) Input-Output  
Prompting (IO)



(c) Chain of Thought  
Prompting (CoT)



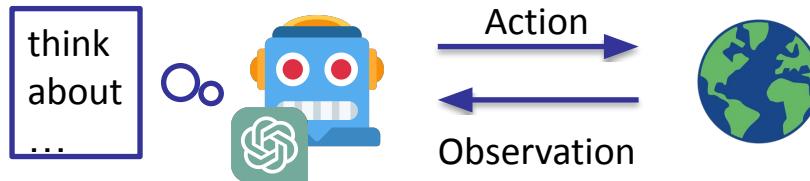
(c) Self Consistency  
with CoT (CoT-SC)



(d) Tree of Thoughts (ToT)

# Reasoning: Takeaways

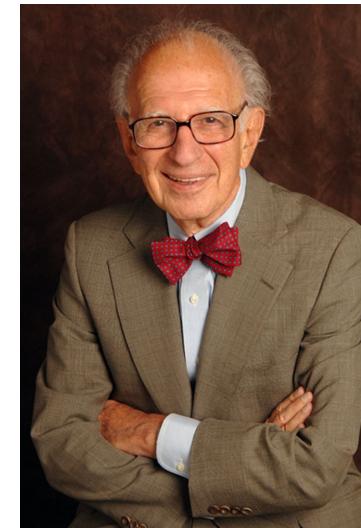
- Reasoning as internal actions for agents
  - No external feedback, just change internal context
  - Infinite space -> requires strong priors to navigate
- Reasoning guides acting, acting updates reasoning
- Action planning approaches (e.g. tree search) can improve reasoning



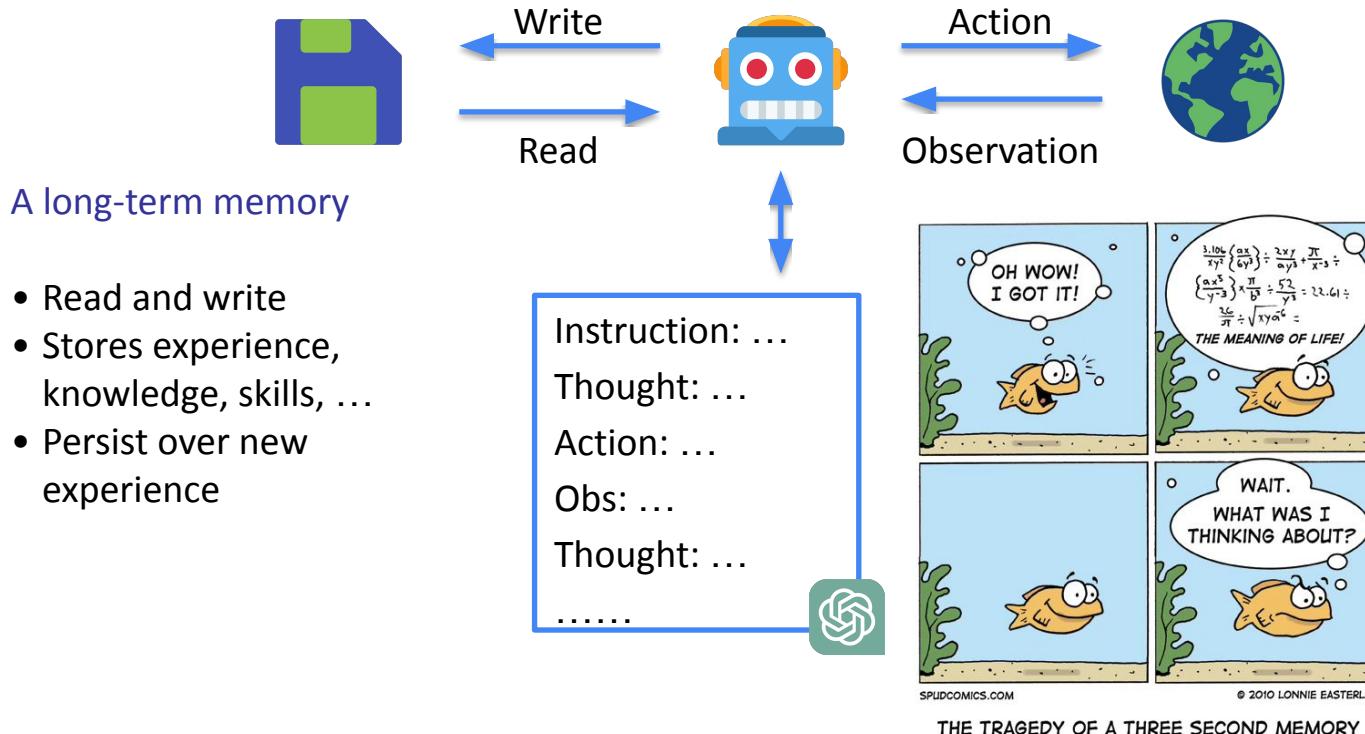
# Memory

Memory is everything. Without it  
we are nothing.

— Eric Kandel

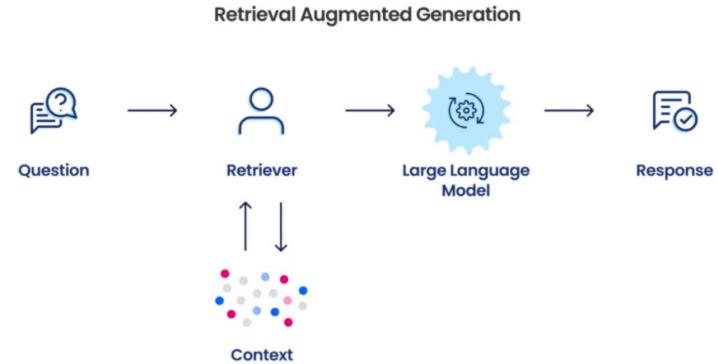






# What about retrieval and RAG?

- We can think of the retrieval corpus as “read-only” LTM
  - Written by others (e.g., Wikipedia editors), not the agent itself
  - Retrieve[query]: a read action
- Limitations
  - Can only live “others’ experience”, which might not be optimal for the agent
  - The way corpus is written might not be optimal for agent usage
- Agent memory: also be able to autonomously write to it!



# Long-term memory: Content

Type by content	Definition	Examples
Episodic memory	Stores experience	Generative agents [Park et al., 2023]
Semantic memory	Stores knowledge	
Procedural memory	Stores skills	Voyager [Wang et al., 2023]

- Note: here we categorize based on memory content, which is
  - Inspired by human long-term memory systems
  - orthogonal to implementations

# Generative agents

## Morning routine



## Catching up



## Beginning workday



## Packing



6:00 am

• • •

7:30 am

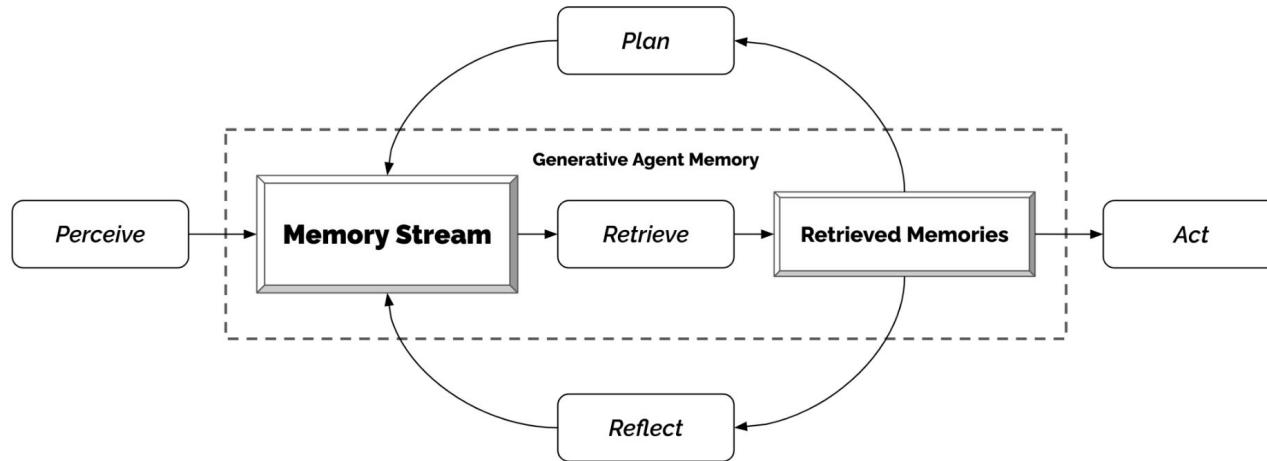
7:45 am

8:00 am



# Generative agents

- The need for memory
  - Context window cannot possibly hold all the event streams
  - Even if possible, might be hard to attend to relevant events, or digest over them



# Episodic memory

- Write: append-only event streams
- Read: retrieval based on heuristic scores

Memory Stream	
2023-02-13 22:48:20:	desk is idle
2023-02-13 22:48:20:	bed is idle
2023-02-13 22:48:10:	closet is idle
2023-02-13 22:48:10:	refrigerator is idle
2023-02-13 22:48:10:	Isabella Rodriguez is stretching
2023-02-13 22:33:30:	shelf is idle
2023-02-13 22:33:30:	desk is neat and organized
2023-02-13 22:33:10:	Isabella Rodriguez is writing in her journal
2023-02-13 22:18:10:	desk is idle
2023-02-13 22:18:10:	Isabella Rodriguez is taking a break
2023-02-13 21:49:00:	bed is idle
2023-02-13 21:48:50:	Isabella Rodriguez is cleaning up the kitchen
2023-02-13 21:48:50:	refrigerator is idle
2023-02-13 21:48:50:	bed is being used
2023-02-13 21:48:10:	shelf is idle
2023-02-13 21:48:10:	Isabella Rodriguez is watching a movie
2023-02-13 21:19:10:	shelf is organized and tidy
2023-02-13 21:18:10:	desk is idle
2023-02-13 21:18:10:	Isabella Rodriguez is reading a book
2023-02-13 21:03:40:	bed is idle
2023-02-13 21:03:30:	refrigerator is idle
2023-02-13 21:03:30:	desk is in use with a laptop and some papers on it
...	



**Q. What are you looking forward to the most right now?**

Isabella Rodriguez is excited to be planning a Valentine's Day party at Hobbs Cafe on February 14th from 5pm and is eager to invite everyone to attend the party.

retrieval	=	recency	importance	relevance
2.34	=	0.91	+ 0.63	+ 0.80

ordering decorations for the party

2.21	=	0.87	+ 0.63	+ 0.71
------	---	------	--------	--------

researching ideas for the party

2.20	=	0.85	+ 0.73	+ 0.62
------	---	------	--------	--------

...

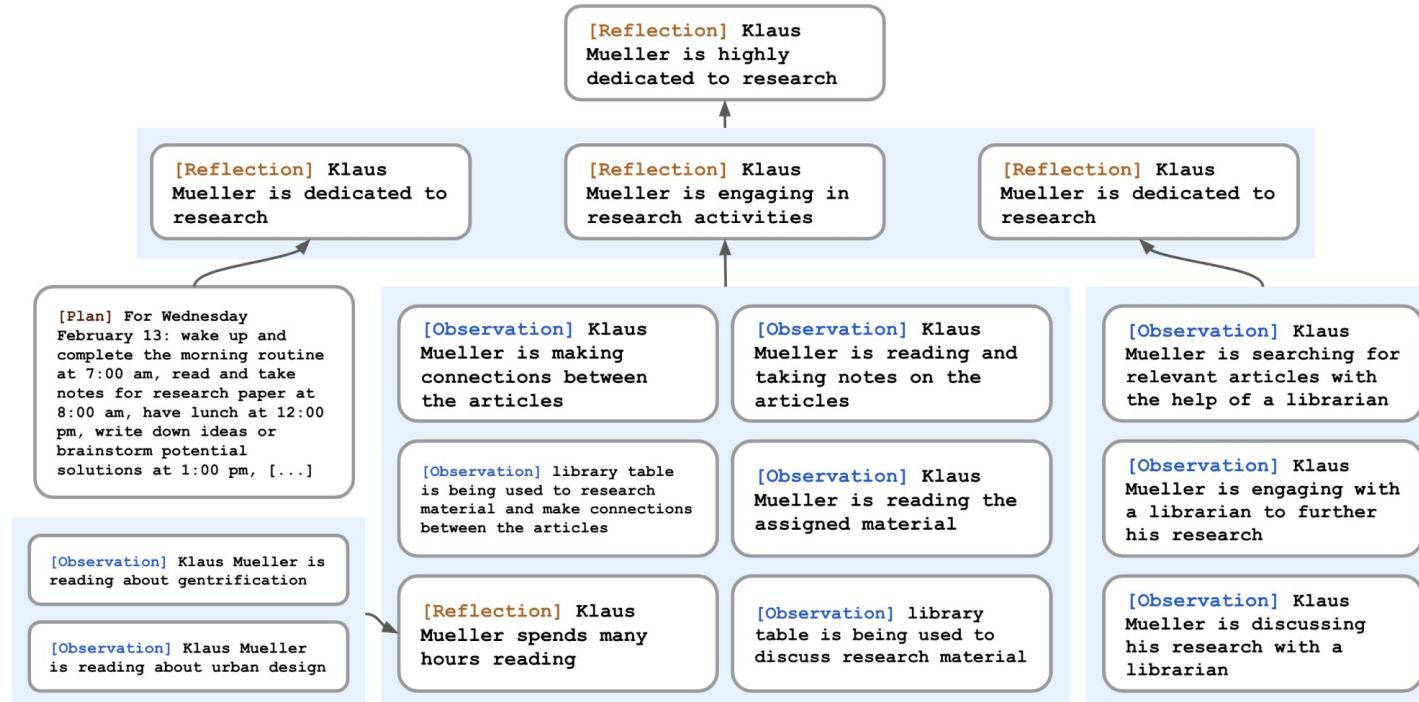


I'm looking forward to the Valentine's Day party that I'm planning at Hobbs Cafe!



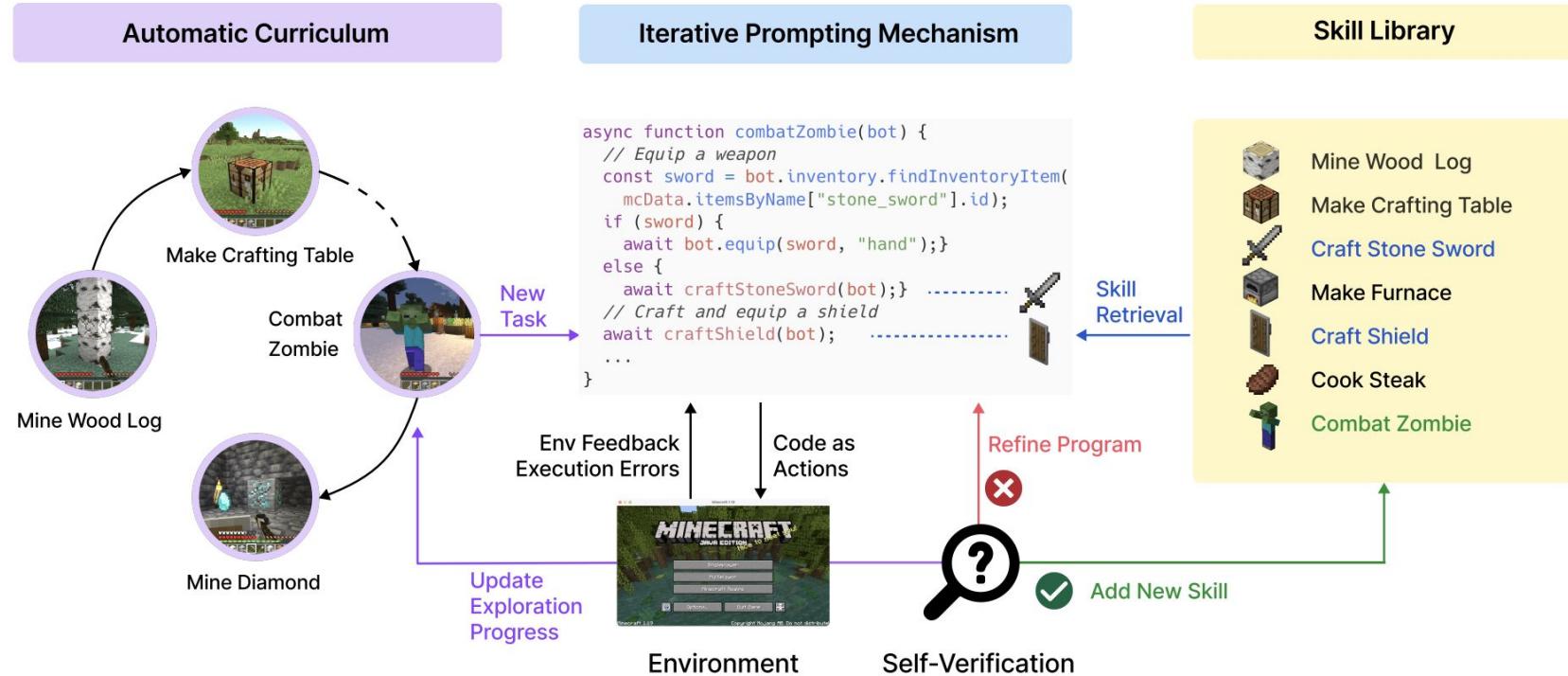
# Semantic memory

- Write: LLM reasoning over events
- Read: retrieval



# Voyager: Procedural memory

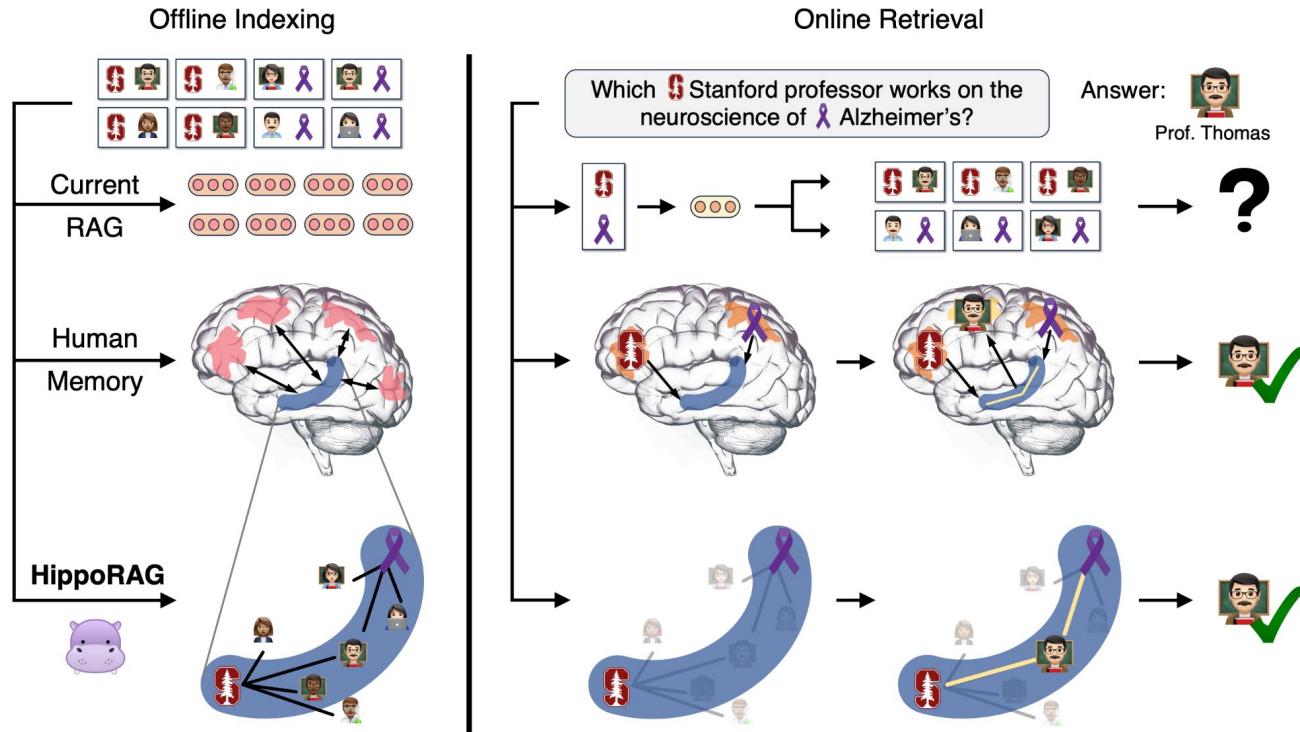
- Write: Code-based skills
- Read: Embedding retrieval



# Long-term memory: Implementation

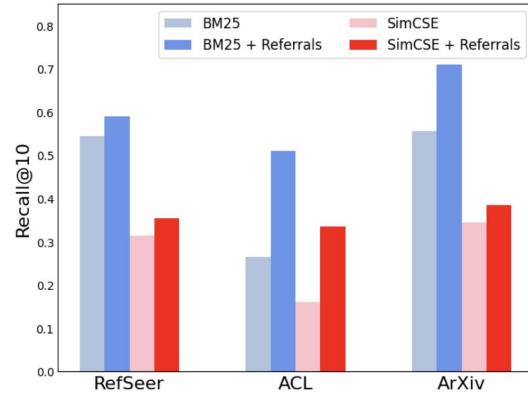
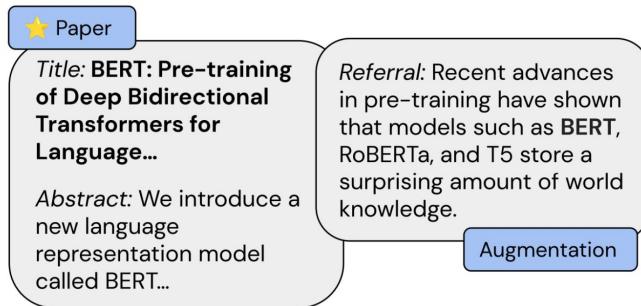
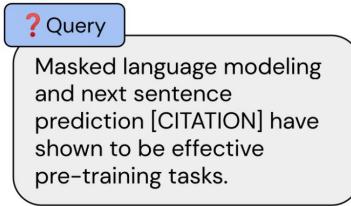
- The “naive” implementation
  - Read: off-the-shelf retriever, e.g. BM25, neural embeddings
  - Write: append to some text corpus
- Some interesting new implementations for agent memory
  - Read: online retrieval by traversal
  - Write: index augmentation with reasoning/referral

# HippoRAG



# Referral augmentation

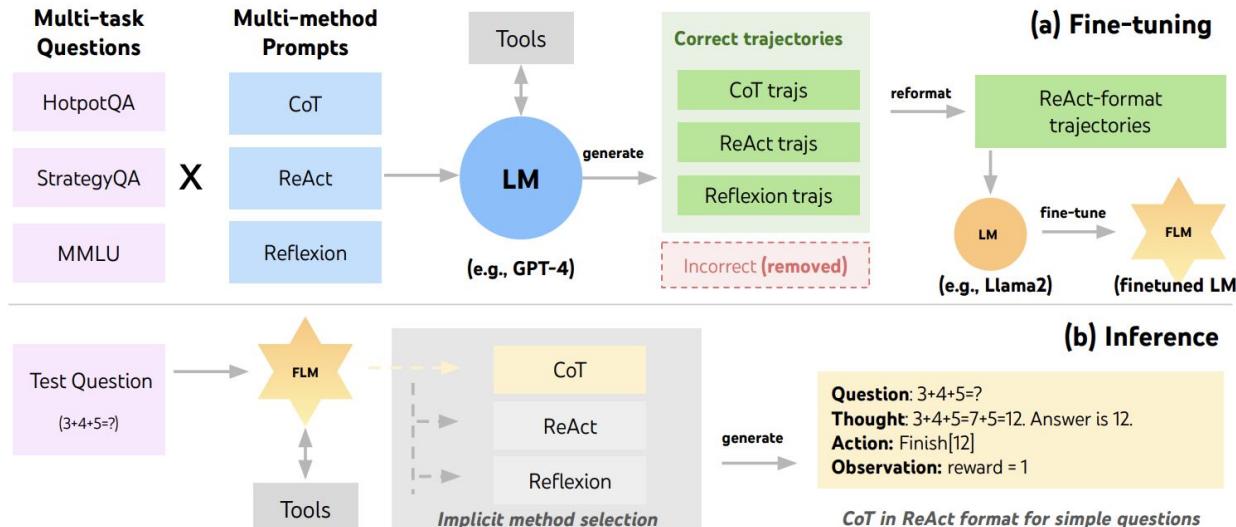
## Paper Retrieval



- Reasoning can be used to augment what's written to memory
- More efforts spent into writing, less spend into reading**

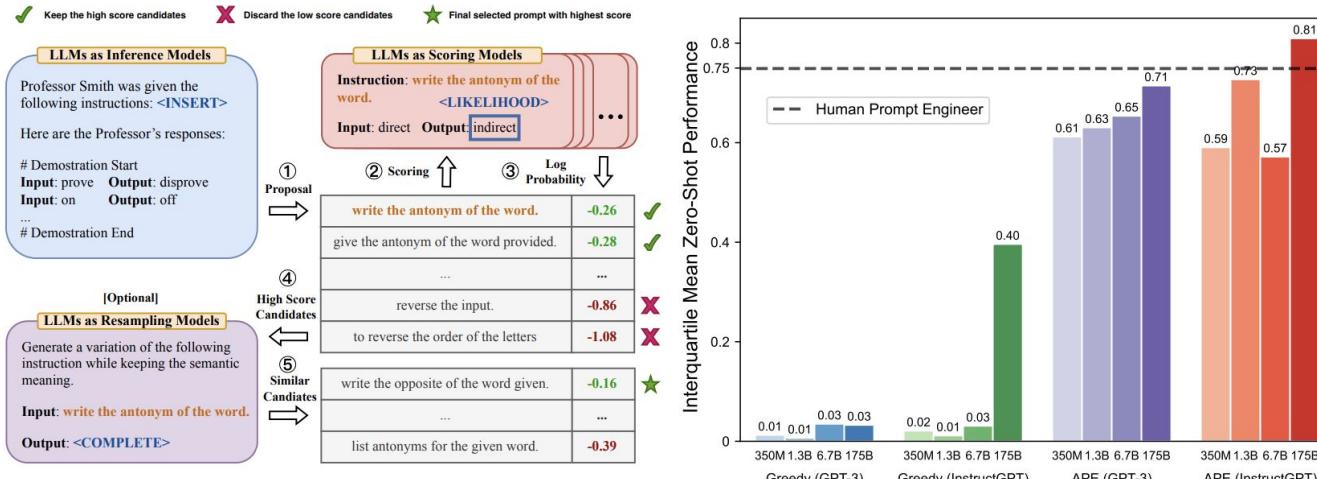
# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)



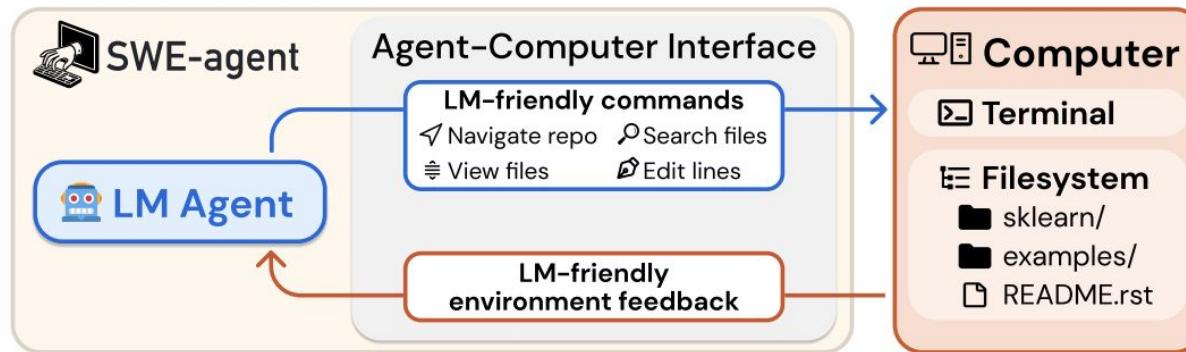
# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)
- Optimizing prompts across tasks



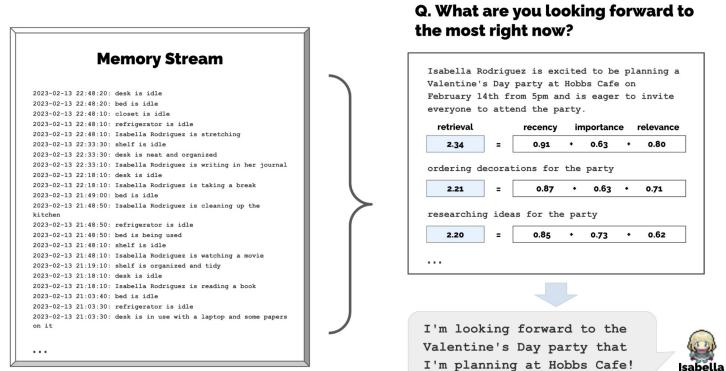
# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)
- Optimizing prompts across tasks
- Improve the agent's own codebase



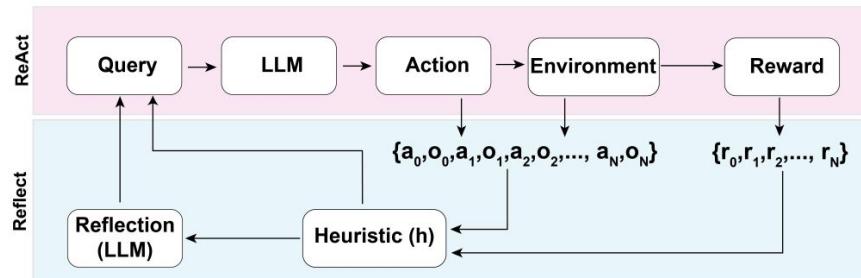
# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)
- Optimizing prompts across tasks
- Improve the agent's own codebase
- Write down examples/events to retrieve later



# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)
- Optimizing prompts across tasks
- Improve the agent's own codebase
- Write down examples/events to retrieve later
- Append self-reflection to prompt to try again



# Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)
- Optimizing prompts across tasks
- Improve the agent's own codebase
- Write down examples/events to remember
- Append self-reflection to prompt
- Combine multiple approaches!

**Fine-Tuning and Prompt Optimization:  
Two Great Steps that Work **Better Together****

Dilara Soylu Christopher Potts Omar Khattab

Stanford University

---

**Algorithm 1 BetterTogether:** Optimizing LM programs by alternating prompt and weight optimization steps, instantiated in **Algorithm 2**

---

**Input:** Program  $\Phi_{\langle\Theta,\Pi\rangle} = \Phi_\Theta \odot \Phi_\Pi$ ,  
with module weights  $\Theta = [\theta_1, \dots, \theta_{|\Phi|}]$   
and module prompts  $\Pi = [\pi_1, \dots, \pi_{|\Phi|}]$

**Training Set  $X$  and Metric  $\mu$**

```
1: function BETTERTOGETHER( $\Phi_{\langle\Theta,\Pi\rangle}$ ,  $X$ ,  $\mu$ )
2:    $\Pi' \leftarrow \text{OPTIMIZEPROMPTS}(\Phi_{\langle\Theta,\Pi\rangle}, X, \mu)$ 
3:    $\Theta' \leftarrow \text{FINETUNEWEIGHTS}(\Phi_{\langle\Theta,\Pi'\rangle}, X, \mu)$ 
4:    $\Pi'' \leftarrow \text{OPTIMIZEPROMPTS}(\Phi_{\langle\Theta',\Pi''\rangle}, X, \mu)$ 
5:   return  $\Phi_{\langle\Theta',\Pi''\rangle}$ 
6: end function
```

---

# Memory: Takeaways

- Language agents interact with external environments and internal memories (information-storing devices)
  - Interact with short-term memory (context window): reasoning
  - Interact with long-term memory (LLM weights, event logs, codebase, prompt library, etc.): retrieving and **learning**
- Exercise question: what's the difference between external environment vs internal memory then? (Hint: check CoALA)

