

Agent Foundations

Reasoning, Memory, and Planning

Slides credit: EMNLP 2024 Tutorial by Yu Su, Diyi Yang, Shunyu Yao and Tao Yu.

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

Reasoning

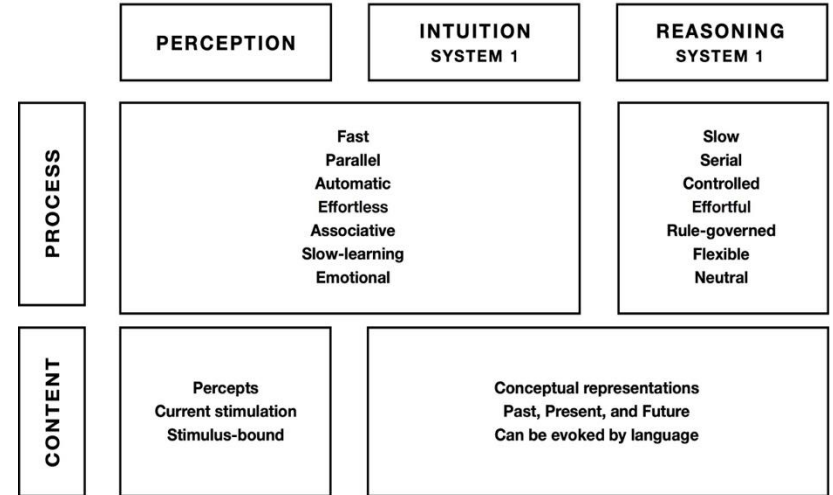
THINK

— Thomas J. Watson



Reasoning

- For humans: various mental processes



*(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^3 , 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^3 , which is less than water. Thus, a pear would float. So the answer is no.

Date Understanding

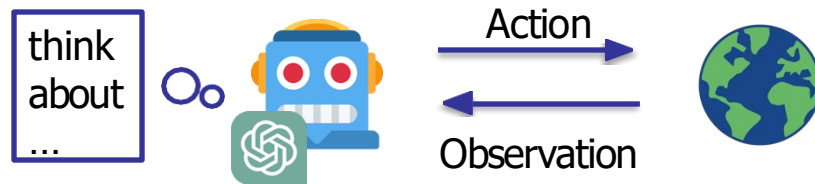
Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

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)

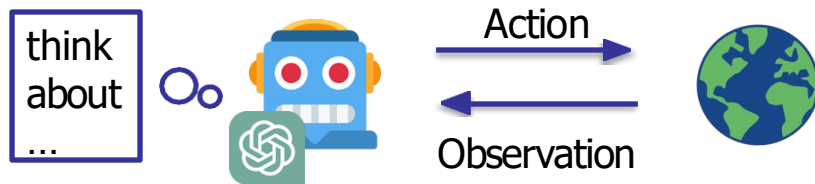
Reasoning

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



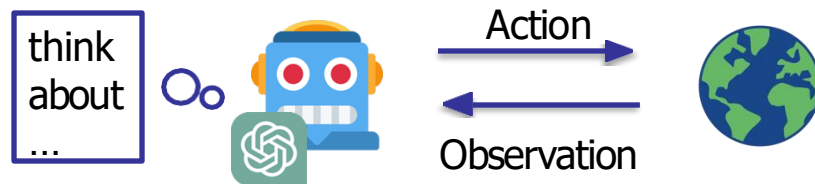
Reasoning

- 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?



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

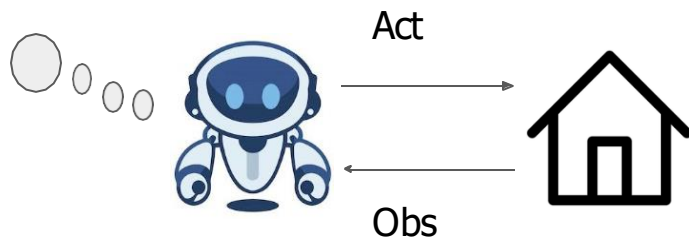
Obs
 $t+1$

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?

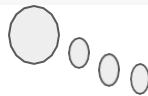


As of December 2022, the Prime Minister of the United Kingdom is Boris Johnson. However, since I am a large language model trained by OpenAI, my knowledge is limited to what was available at the time of my training in 2021. My ability to provide current information is limited, and I am not able to browse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

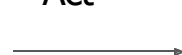
Act to reason

1. Obtain knowledge
2. Update knowledge

Hallucination due to limited / outdated knowledge



Act



Obs



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

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]



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.

Fewer Actions Required!

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)
```

Control & Data Flow of Code Simplifies Complex Operations

 **Environment** 1.1, 0.19

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

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.

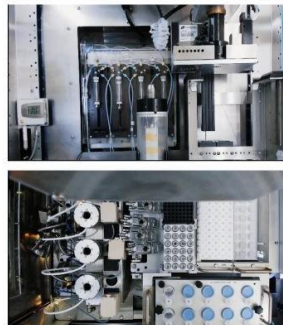
SynthesisPlanner tool: Detailed synthesis plan

I will execute the synthesis.

SynthesisExecutor tool: Successful synthesis.

c

RoboRXN synthesis platform

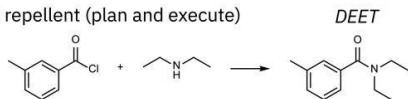


Connection with
physical world



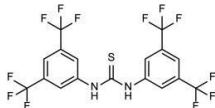
d Chemcrow workflows with experimental validation

Insect repellent (plan and execute)

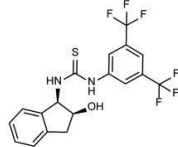


Thiourea organocatalysts (plan and execute)

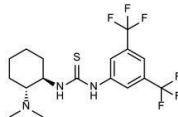
Schreiner's catalyst



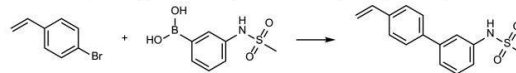
Ricci's catalyst



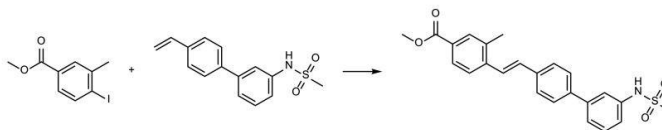
Takemoto's catalyst



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...

→ 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".

MLAgentBench,
2023

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

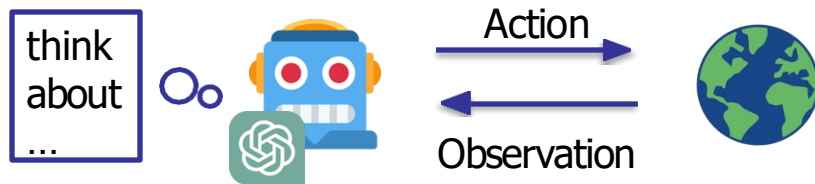
Speak in fictional language

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

ORIBA, 2023

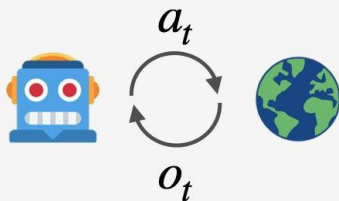
Reasoning

- 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?



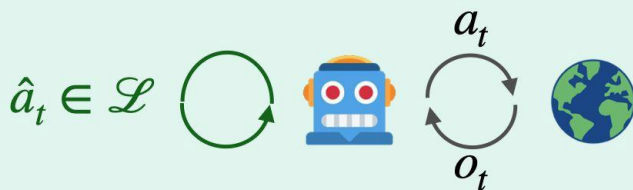
Why is reasoning special for agents?

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 $a_t \sim \pi(a | c_t) \in A$

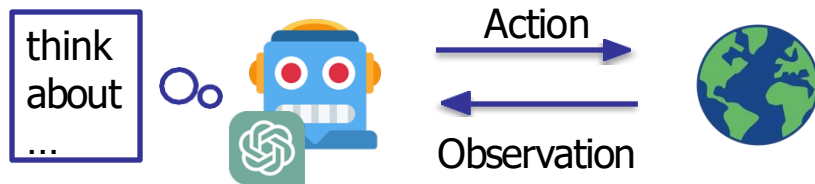
ReAct: action space $\hat{A} = A \cup \mathcal{L}$ augmented by reasoning



- $\hat{a}_t \in \mathcal{L}$ 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**

Reasoning

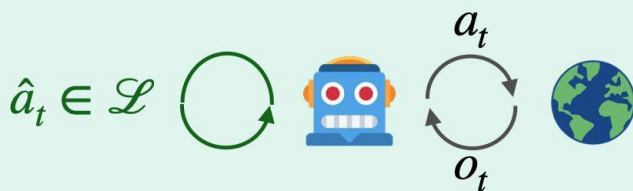
- For humans: various mental processes
- For LMs: intermediate generation
- For agents: internal actions
 - Why is it helpful?
 - Why is it special?
 - Why is it now?



Why is reasoning just now for agents?

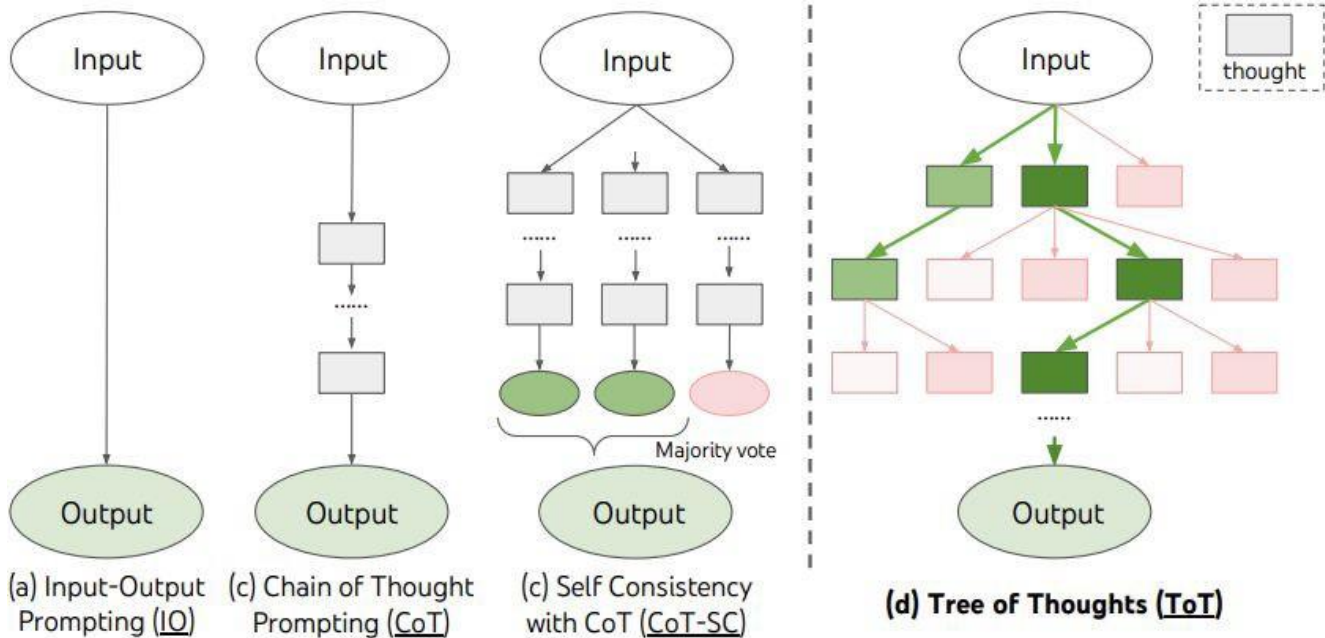
- Bigger action space -> More capacity, harder decision making
 - The space of reasoning/language is **infinite**
- LLMs learn reasoning priors by imitating various human reasoning traces

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



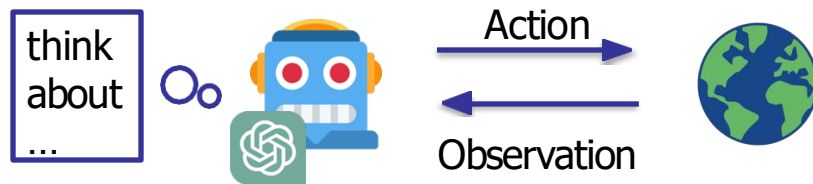
- $\hat{a}_t \in \mathcal{L}$ can be any language sequence
- Agent context $c_{t+1} = (c_t, \hat{a}_t, a_t, o_{t+1})$
- $\hat{a}_t \in \mathcal{L}$ only updates **internal context**

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



Reasoning: Takeaways

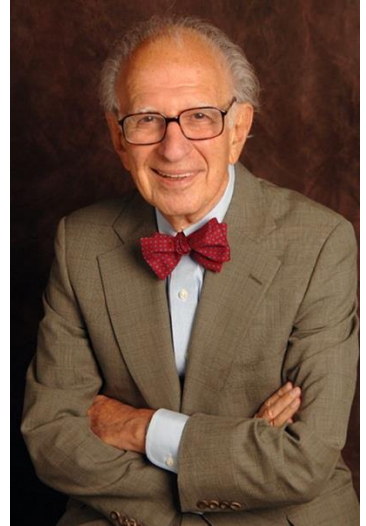
- Reasoning as internal actions for agents
 - No external feedback, just change internal context
 - Infinite space -> requires strong priors to navigate
- Reasoning guides acting, acting updates reasoning
- Action planning approaches (e.g. tree search) can improve reasoning

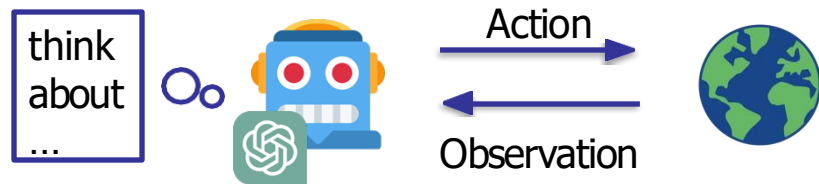


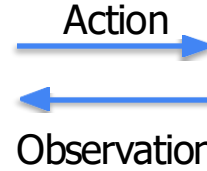
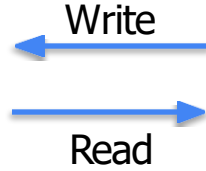
Memory

Memory is everything. Without it
we are nothing.

— Eric Kandel



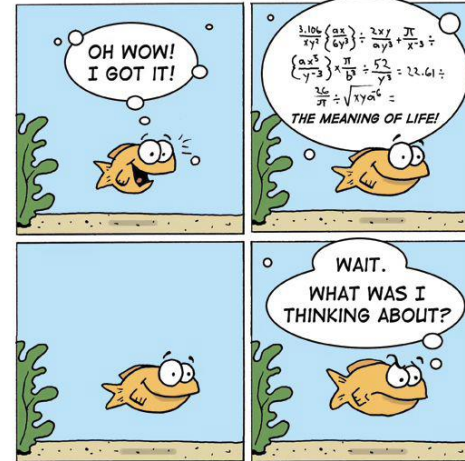




A long-term memory

- Read and write
- Stores experience, knowledge, skills, ...
- Persist over new experience

Instruction: ...
Thought: ...
Action: ...
Obs: ...
Thought: ...
... ..



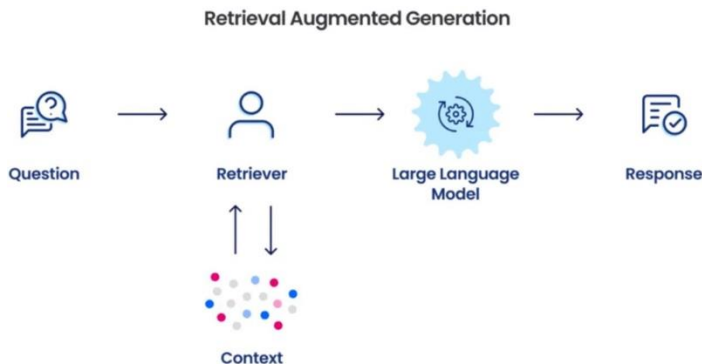
SPUDCOMICS.COM

© 2010 LONNIE EASTERLING

THE TRAGEDY OF A THREE SECOND MEMORY

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



Waking up



Brushing teeth



Taking a shower



Cooking breakfast

Catching up



Packing



Beginning workday



6:00 am

■ ■ ■

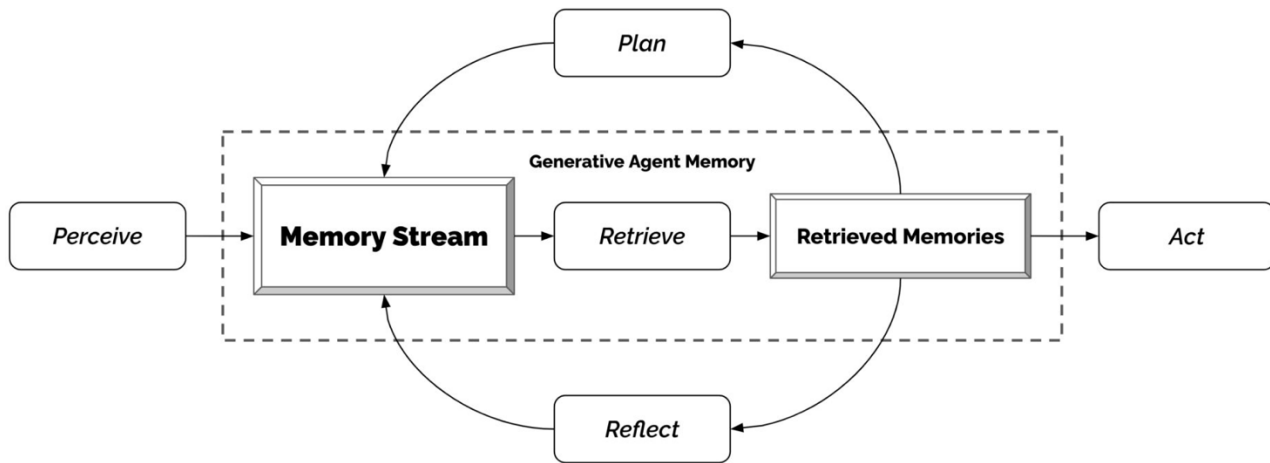
7:30 am

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
------	---	------	--------	--------

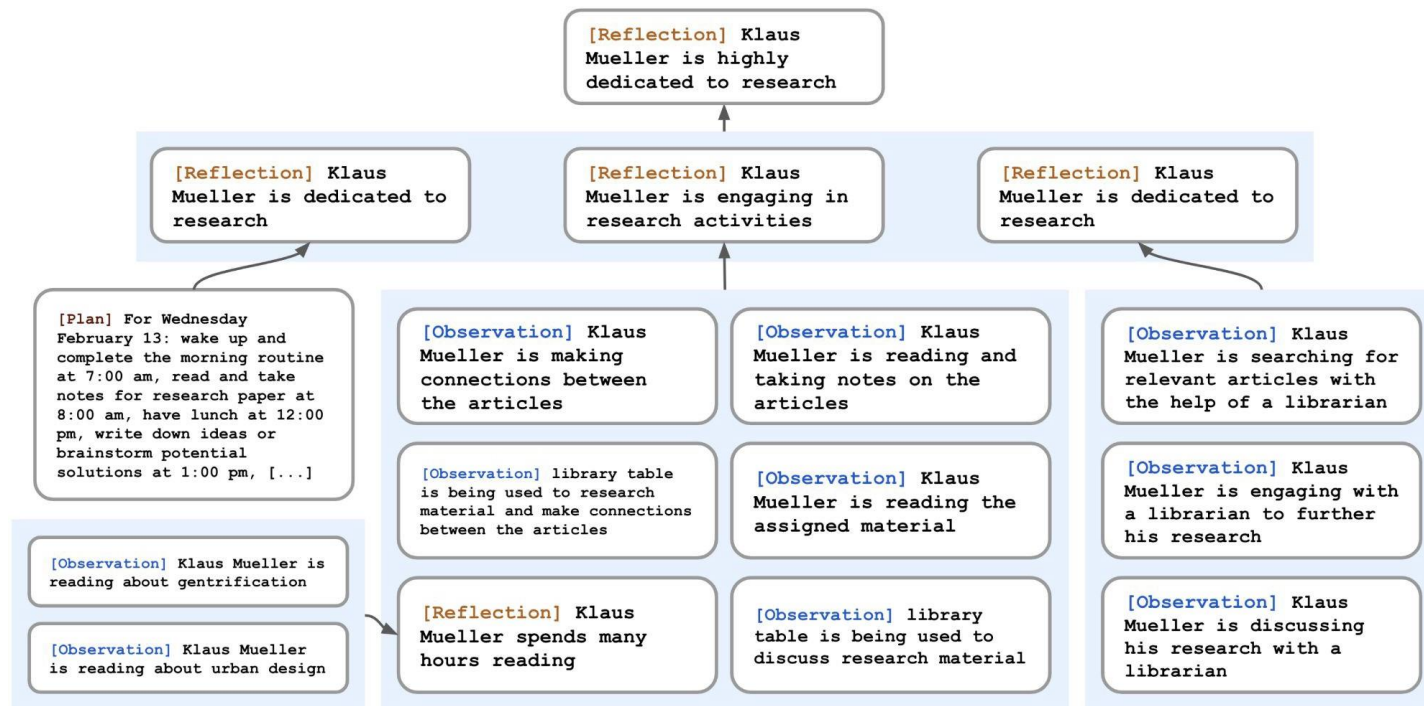
...

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



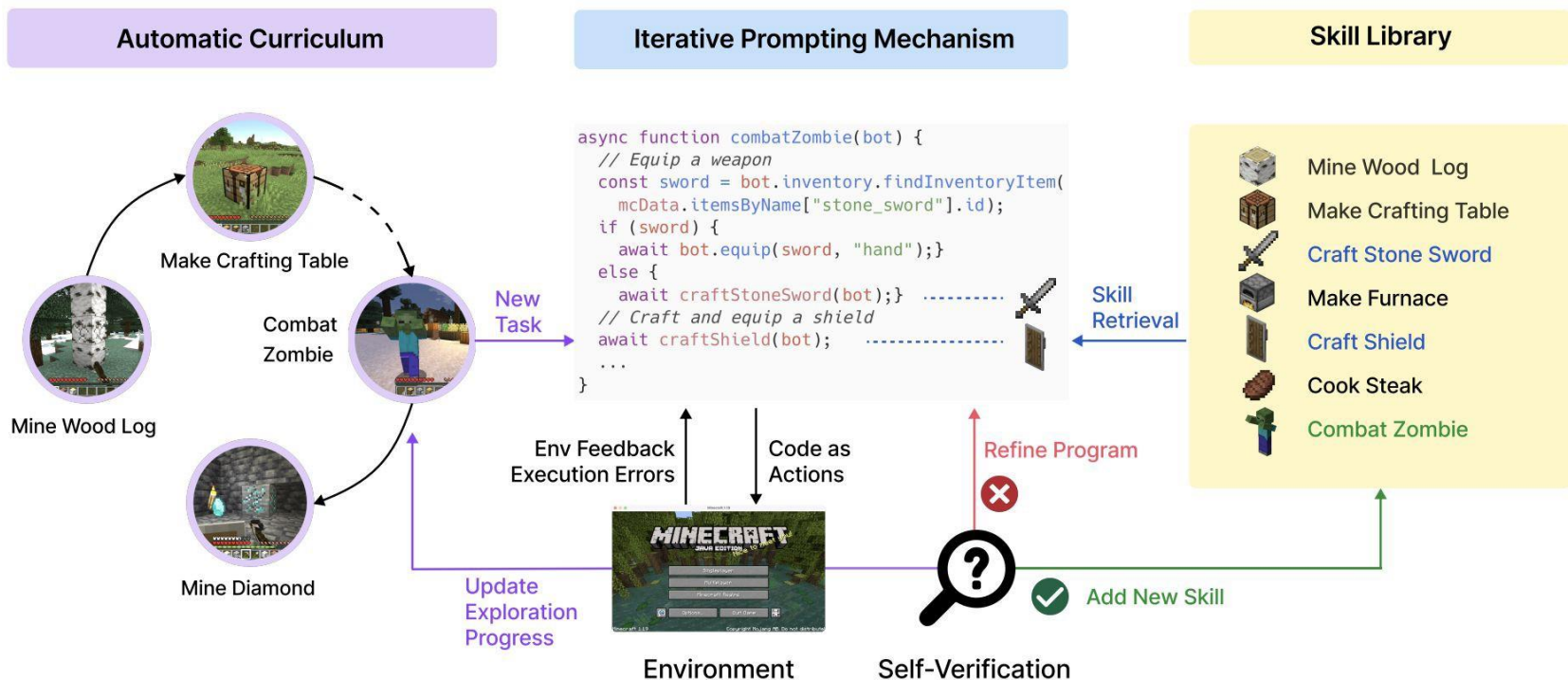
Semantic memory

- Write: LLM reasoning over events
- Read: retrieval



Voyager: Procedural memory

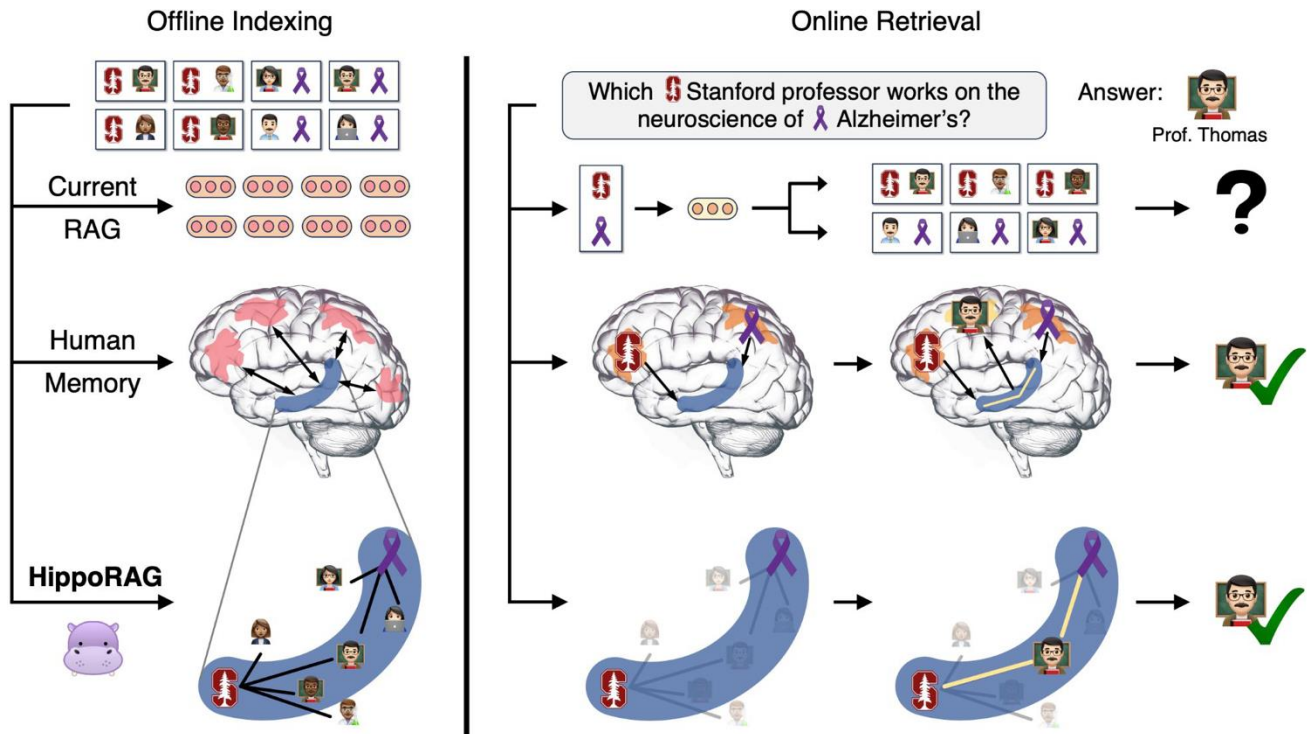
- Write: Code-based skills
- Read: Embedding retrieval



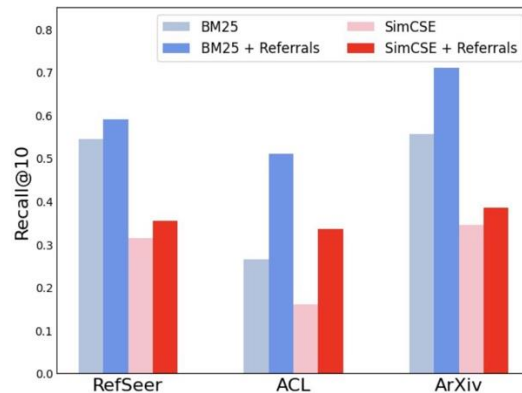
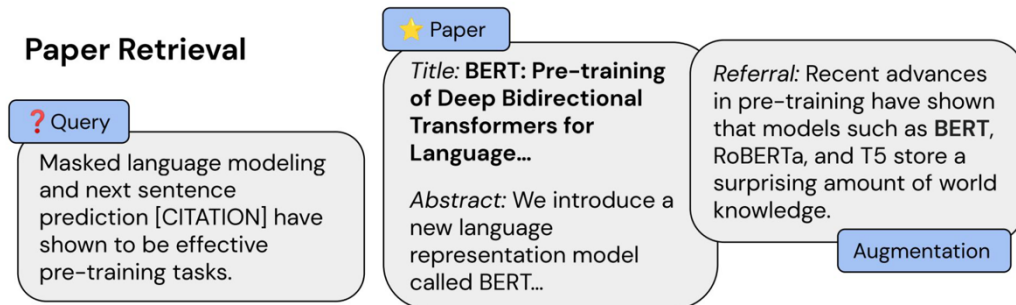
Long-term memory: Implementation

- 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



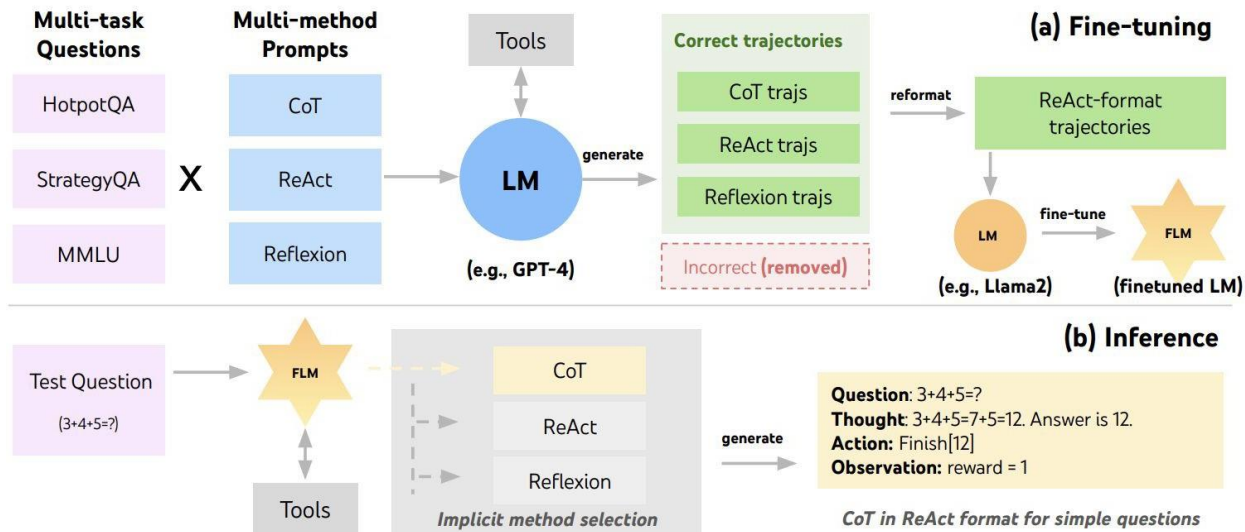
Referral augmentation



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

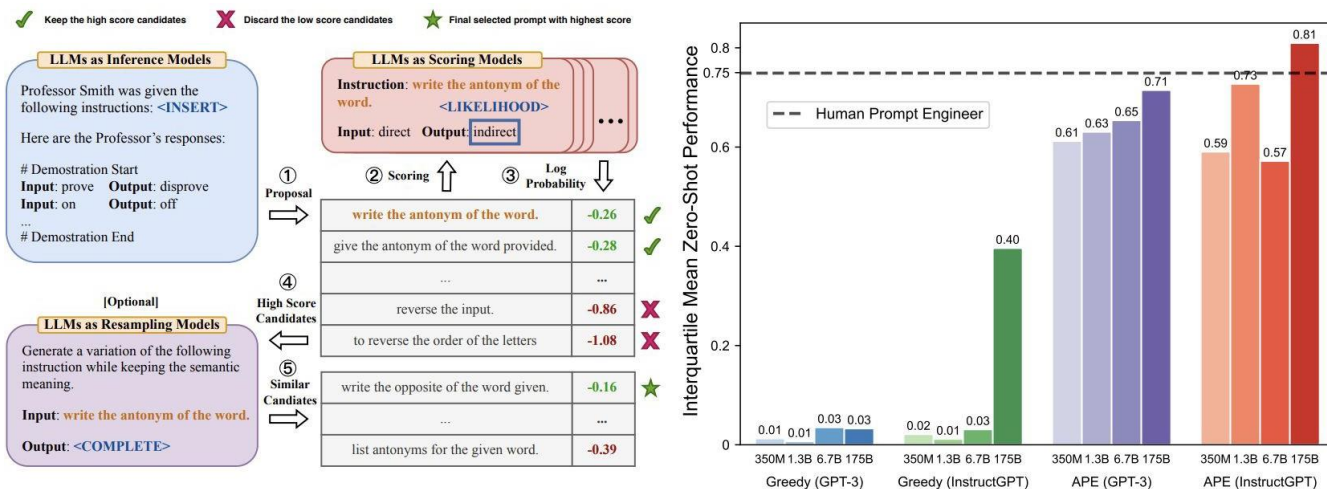
Long-term memory: A unified view of learning

- Fine-tuning model weights (the most obvious)



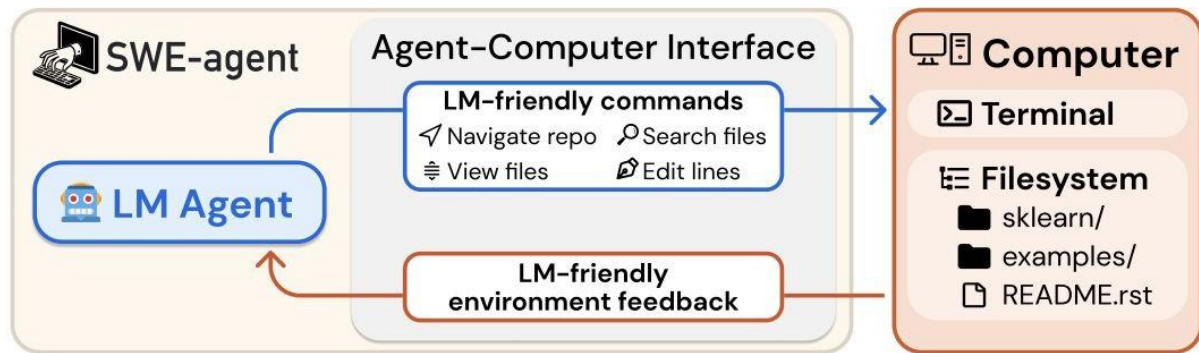
Long-term memory: A unified view of learning

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



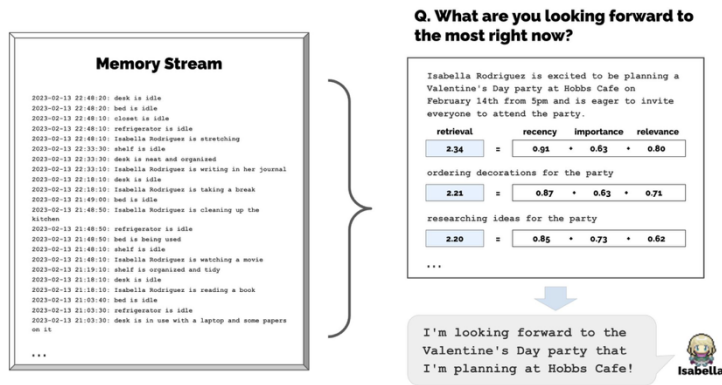
Long-term memory: A unified view of learning

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



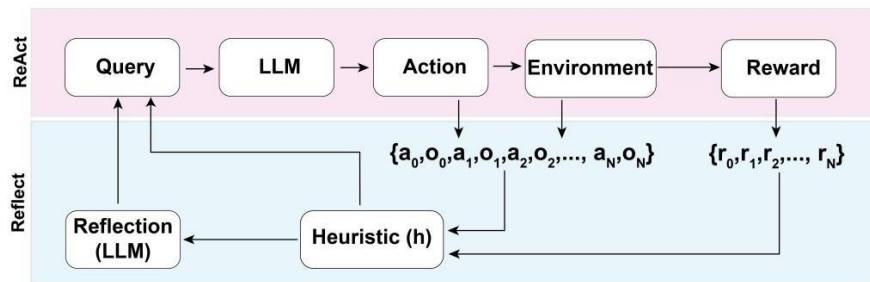
Long-term memory: A unified view of learning

- 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 r
- Append self-reflection to prompt t
- 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 μ

- 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)



Planning



Planning: (simplified) definition

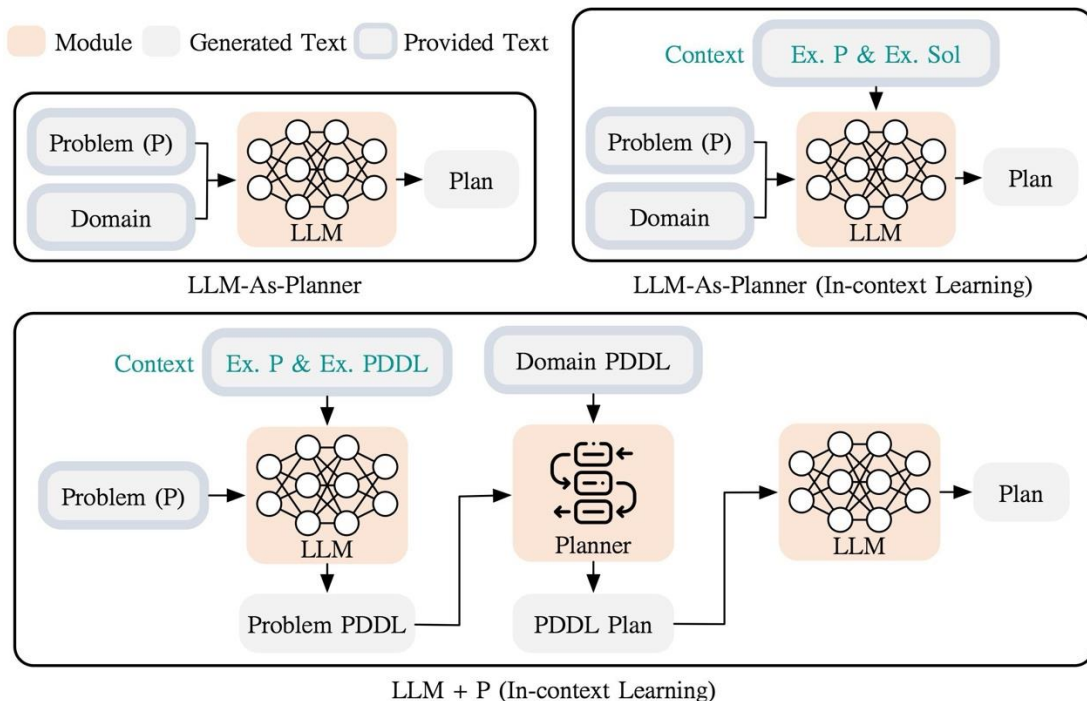
Given a goal G , decides on a sequence of actions $(a_0, a_1, \dots, a_n)$ that will lead to a state that passes the goal test $g(\cdot)$

General trends in planning settings for language agents

- Increasing expressiveness in **goal specification**, e.g., in natural language as opposed to formal language
- Substantially expanded or open-ended **action space**
- Increasing difficulty in automated **goal test**

LLM+P

When domain and problem can be (fully or partially) described formally, e.g., in PDDL (Planning Domain Definition Language)



An Example PDDL Problem File Written by GPT-4 with Context

Context (shaded):

```
An example planning problem is: You have 5 blocks.
b2 is on top of b5. b5 is on top of b1. b1 is on top of
b4. b3 is on top of b2. b4 is on the table. b3 is clear.
Your arm is empty. Your goal is to move the blocks.
b4 should be on top of b3. The problem PDDL is:
(:objects b1 b2 b3 b4 b5)
(:init (arm-empty) (on b1 b4) (on
b2 b5) (on b3 b2)
(on-table b4) (on b5 b1)
(clear b3))
(:goal (and (on b4 b3))))
```

Prompt: Description of Problem (P1) + Context + Provide me with the problem PDDL file that describes the planning problem directly without further explanations.

GPT-4 (the generated problem PDDL):

```
(:objects b1 b2 b3 b4 b5 )
(:init (arm-empty) (on-table b1)
(on b2 b1) (on b3 b4)
(on b4 b2) (on b5 b3) (clear
b5)))
(:goal (and (on b1 b2) (on b3 b5)
(on b4 b1)))
```

Planner:

```
(unstack b5 b3) (putdown b5)
(unstack b3 b4) (stack b3 b5)
(unstack b4 b2) (putdown b4)
(unstack b2 b1) (putdown b2)
(pickup b1) (stack b1 b2) (pickup
b4) (stack b4 b1)
```

Liu et al., "LLM+P: Empowering Large Language Models with Optimal Planning Proficiency." arXiv preprint 2304.11477 (2023)

See also Kambhampati et al., "LLMs Can't Plan, But Can Help Planning in LLM-Modulo Frameworks." arXiv preprint 2402.01817 (2024)

Language agent planning: web agents

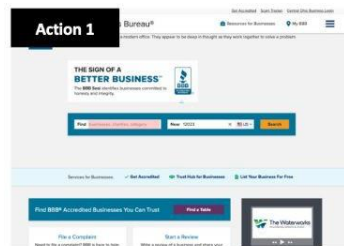
Task Description:

Show me the reviews for the auto repair business closest to 10002.

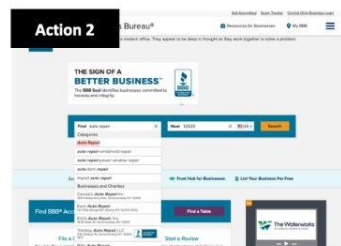
Action Sequence:

Target Element	Operation
1. [searchbox] Find	TYPE: auto repair
2. [button] Auto Repair	CLICK
3. [textbox] Near	TYPE: 10002
4. [button] 10002	CLICK
5. [button] Search	CLICK
6. [switch] Show BBB Accredited only	CLICK
7. [svg]	CLICK
8. [button] Sort By	CLICK
9. [link] Fast Lane 24 Hour Auto Repair	CLICK
10. [link] Read Reviews	CLICK

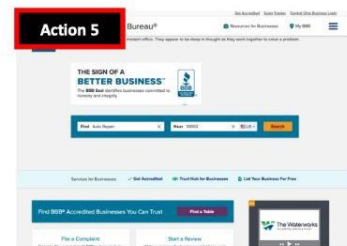
Webpage Snapshots:



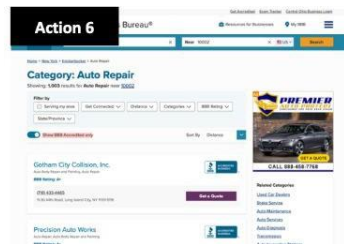
`<input name="find_text" type="search">`



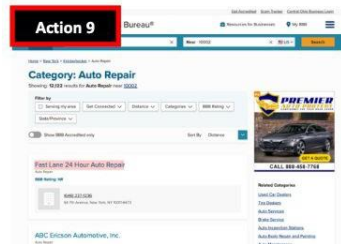
`<em>Auto Repair</em>`



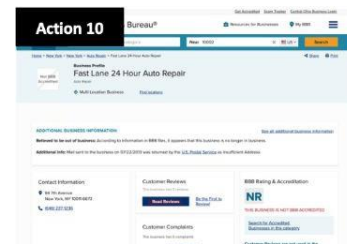
`<button>Search</button>`



`<button>Show BBB Accredited only</button>`

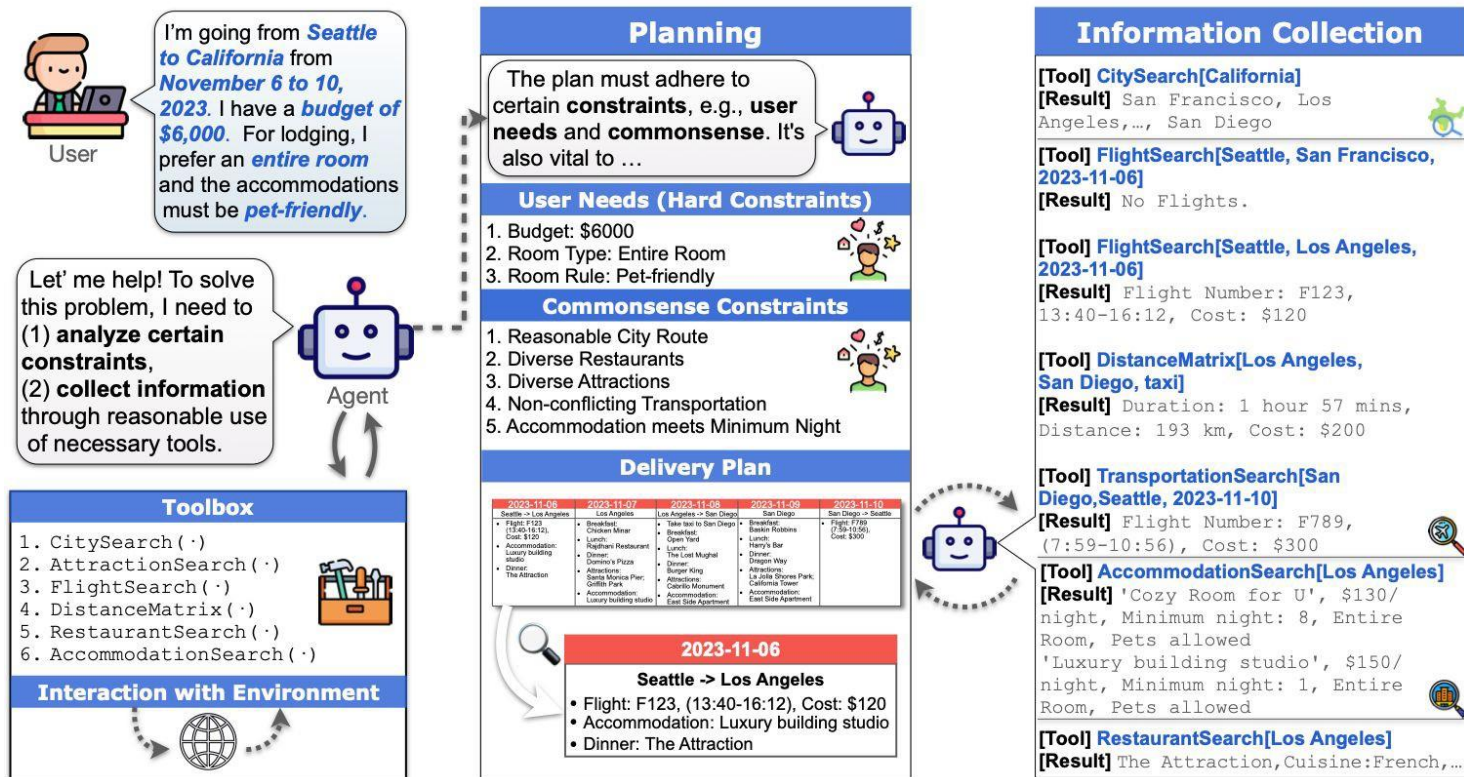


`<span>Fast Lane 24 Hour Auto Repair</span>`



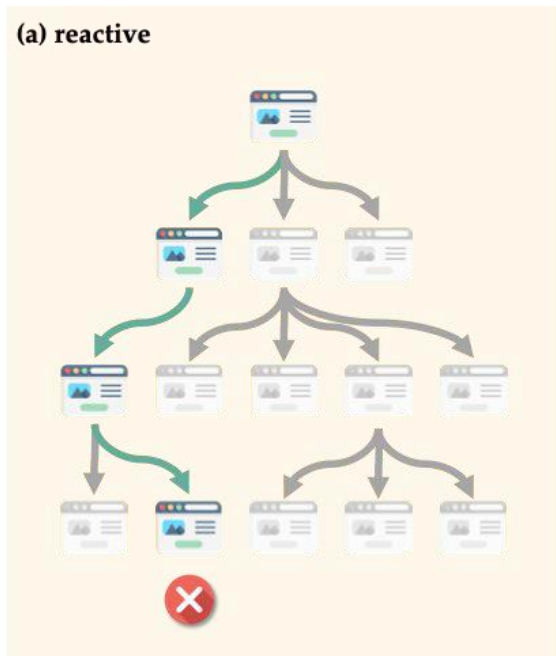
`<a href="link:XXX">Read Reviews</a>`

Language agent planning: travel planning

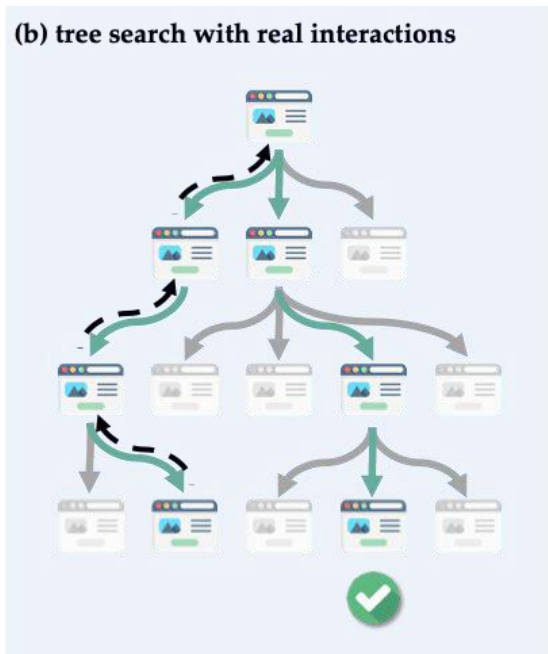


Planning paradigms for language agents

(a) reactive



(b) tree search with real interactions

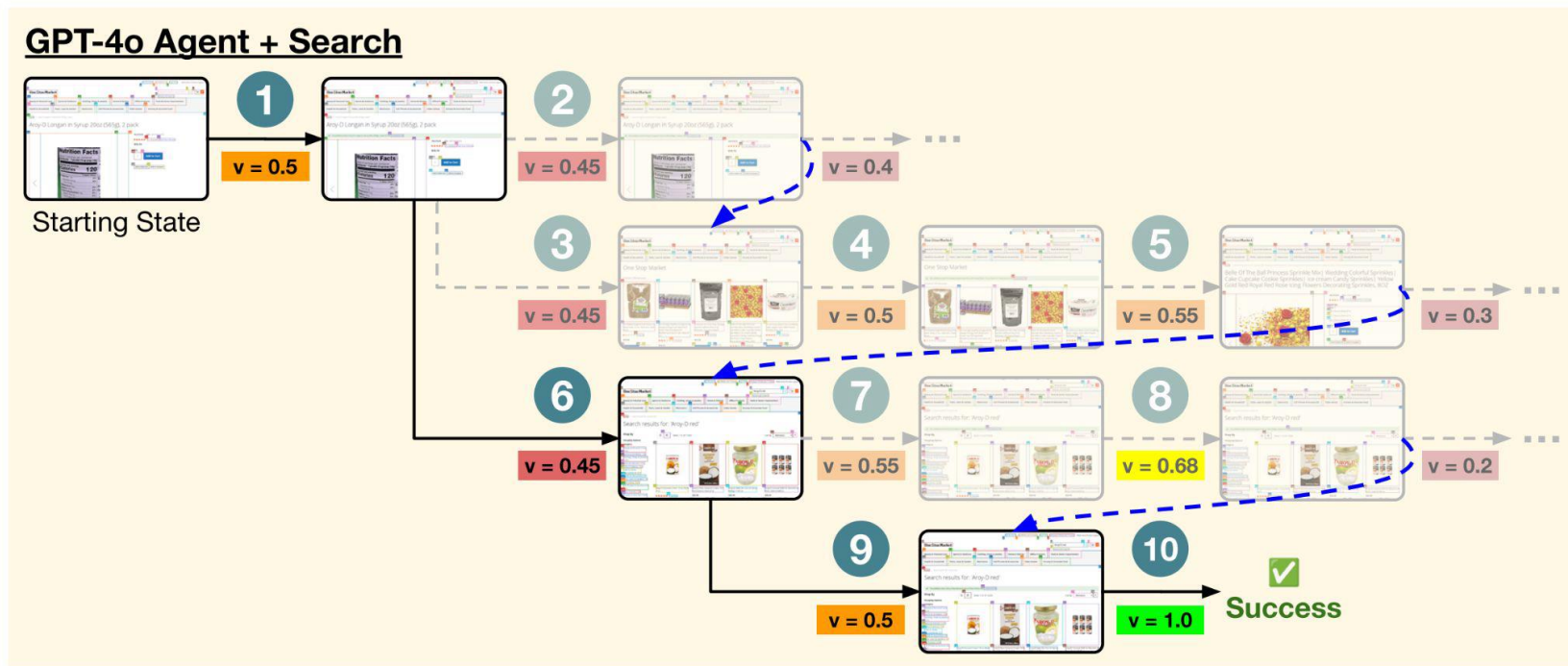


fast, easy to implement



greedy, short-sighted

Tree search with real interactions



Jing Yu Koh, Stephen McAleer, Daniel Fried, Ruslan Salakhutdinov. "Tree Search for Language Model Agents." arXiv preprint arXiv:2407.01476 (2024).
Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, Yu-Xiong Wang. "Language Agent Tree Search Unifies Reasoning Acting and Planning in Language Models." ICML (2024).

Challenges with tree search in the real world

- Many actions are state-changing and irreversible → backtracking ❌
- Safety/privacy risks
- Inference-time exploration could be slow and costly

Cancel Place Your Order - Amazon.co...

 Nespresso Capsules Vertuo, Variety Pack, Medium and Dark Roast Coffee, 30 Count Coffee Pods, Brews 7.8 oz.
\$37.50 (\$1.25 / Count)
✓prime
Ships from and sold by Amazon.com

Quantity: 1 Change
Add gift options
Auto-deliver and save up to 5% on future auto-deliveries
Item often ships in manufacturer's container to reduce packaging and reveals what's inside. To change, click below.

Reduce packaging, ship in manufacturer's container

Place your order

By placing your order, you agree to Amazon's [privacy notice](#) and [conditions of use](#).

← Search or ask a question

Kohl's Dropoff FREE

Kohl's will pack, label, and ship your return for free. Just bring the item in its original manufacturer's packaging and disassemble the item (if applicable). We'll email you a QR code to ship your return. Show it to a store associate at any Kohl's store.

[Find a participating Kohl's store](#)
Printer not required.

☐ The UPS Store locations only—no label needed \$6.99

☐ Amazon Dropoff — box and label needed FREE

2 OTHER RETURN OPTIONS

Refund summary \$13.21

Confirm your return

Verify mobile number

A text with a One Time Password (OTP) has been sent to your mobile number: 8058671234 [Change](#)

Enter OTP: [Resend code](#)

Create your Amazon account

By creating an account, you agree to Amazon's [Conditions of Use](#) and [Privacy Notice](#).

← Search or ask a question

Location

Disabled

AddressBook/Checkout

Your current location will be used to assist in adding a new address to your Amazon address book.

Amazon Cash

We use your location to find nearby stores where you can add money to your Amazon balance with Amazon Cash.

Branded Store Experience Location

We use your location to power branded store experiences.

Branded Store Experience Location-based Augmented Reality

We use your camera, motion, and location to power branded store experiences. Requires camera, motion, and location.

Campus pickup

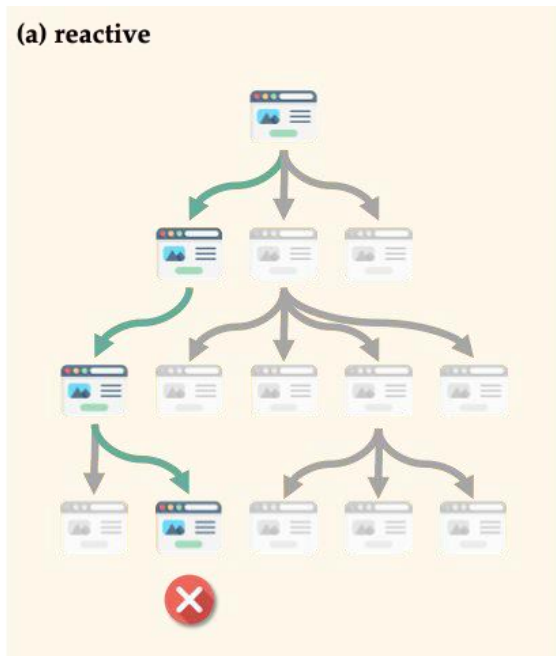
We'll use your location to show the nearby pickup points

Delivery Location

We use your location to improve your shopping experience, ensuring you only see products and delivery options available in your area.

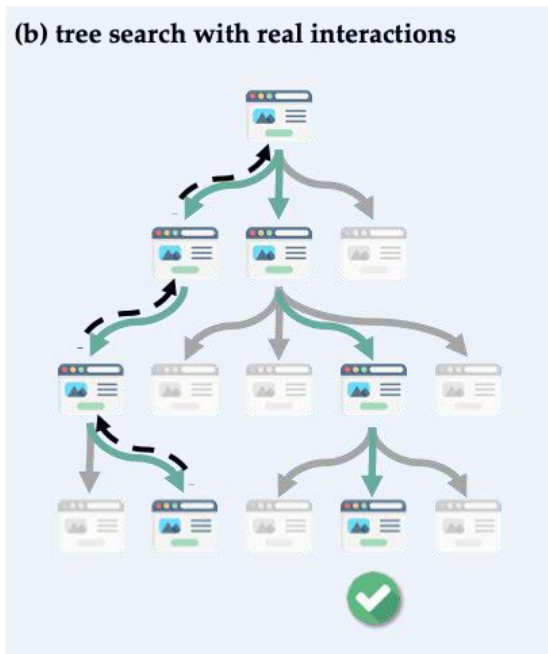
Planning paradigms for language agents

(a) reactive



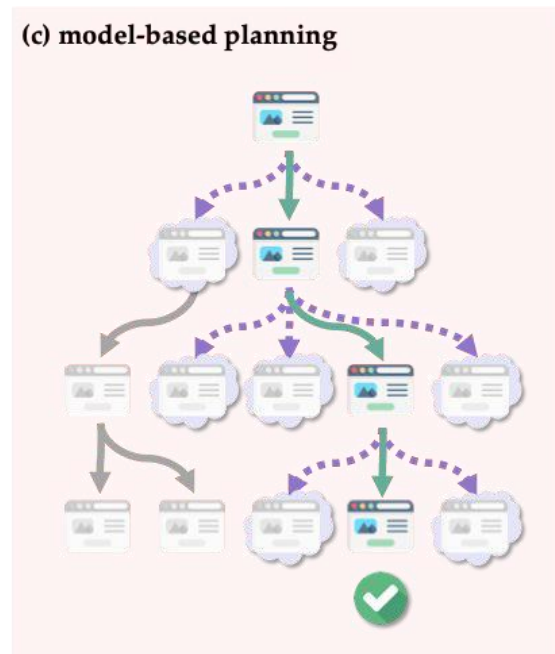
- 😊 fast, easy to implement
- 😞 greedy, short-sighted

(b) tree search with real interactions



- 😊 systematic exploration
- 😞 irreversible actions, unsafe, slow

(c) model-based planning



- 😊 faster, safer, systematic exploration
- 😞 how to get a world model?

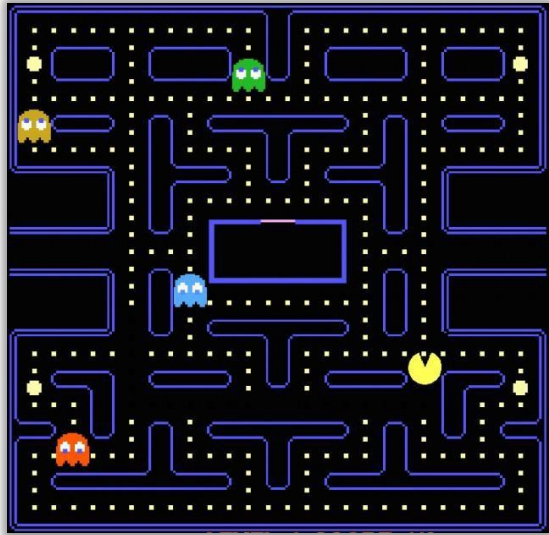
What's ... a world model?

A computational model of environment transition dynamics

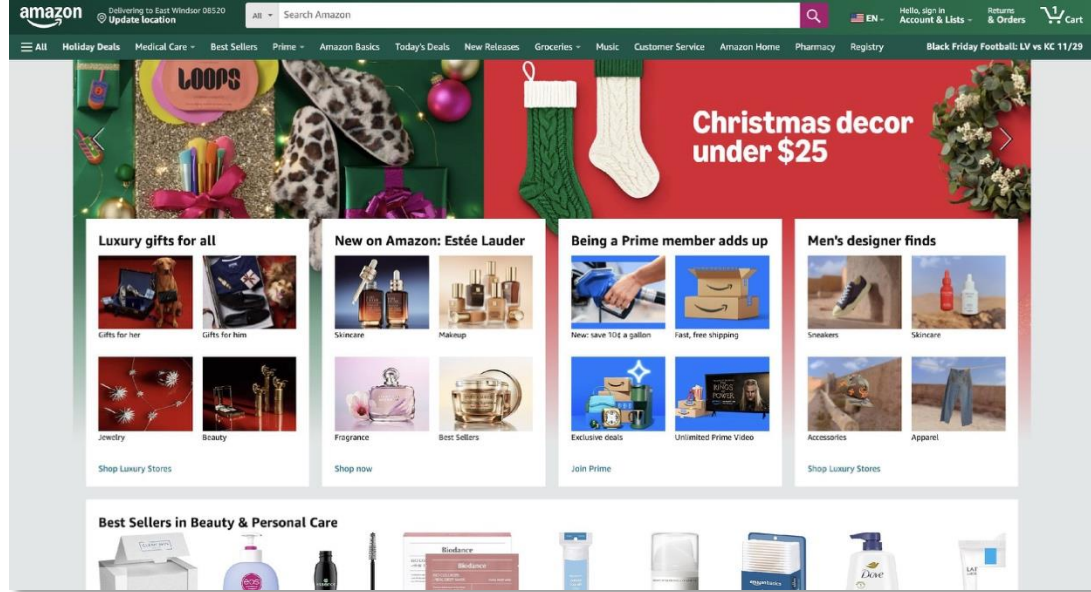
$$\hat{T}: \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$$

If I do this (a_t) right now (s_t), what would happen next (s_{t+1})?

Why hasn't it been done yet?

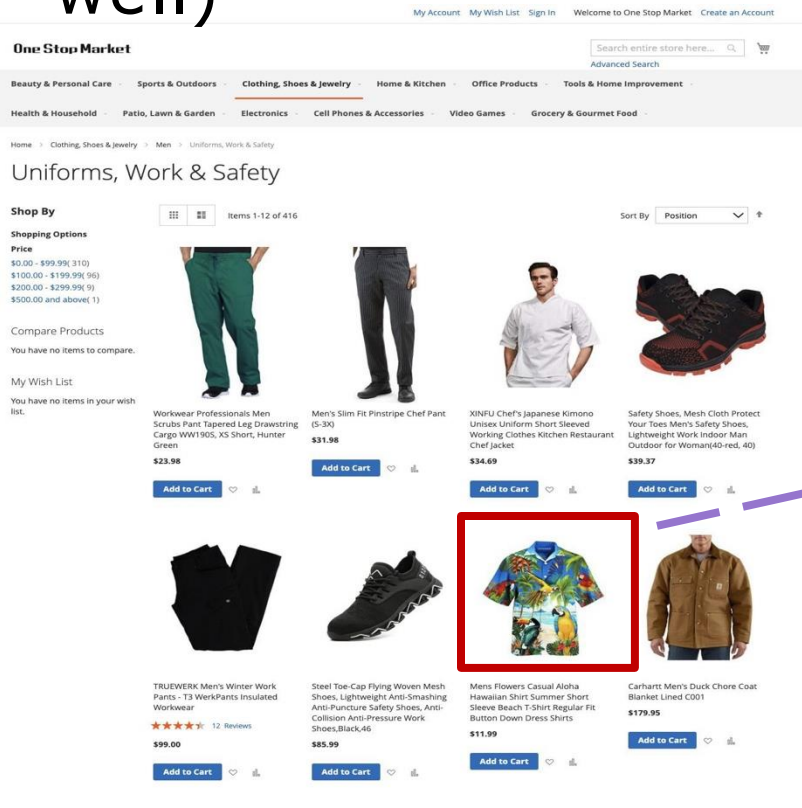


V.S



And **billions of other websites** on the Internet!

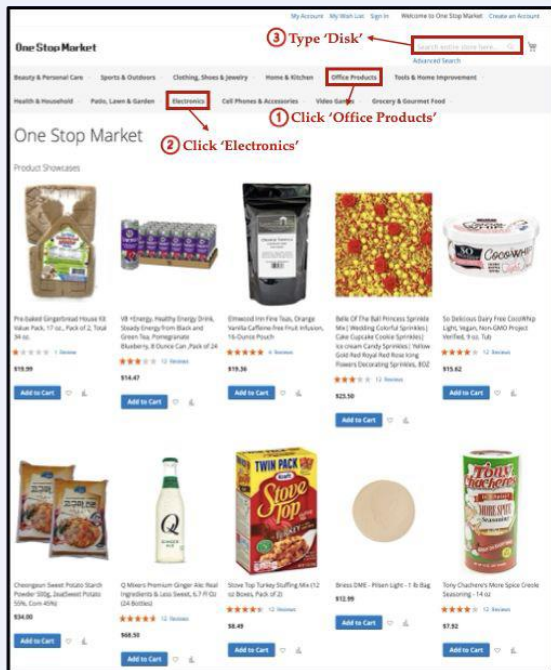
LLMs can predict state transitions (reasonably well)



The page will navigate to a detailed product page for the "Mens Flowers Casual Aloha Hawaiian Shirt Summer Short Sleeve Beach T-Shirt Regular Fit Button Down Dress Shirts." This new page will likely contain additional information about the product including more detailed specifications, customer reviews, larger images, sizing options, and possibly a larger "Add to Cart" button. Other elements from the current category view like the grid of products will be replaced with the detailed view of this specific product.

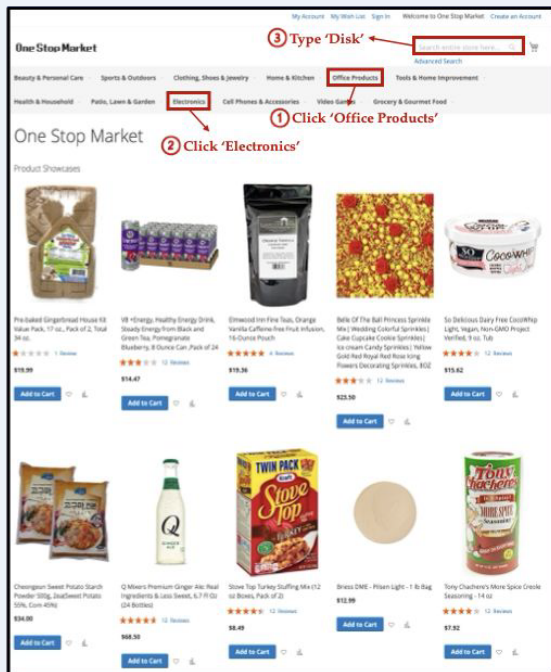
WebDreamer: model-based planner

Please navigate to the 'Data Storage' category and purchase the least expensive disk with 512GB of storage.



WebDreamer: model-based planner

Please navigate to the 'Data Storage' category and purchase the least expensive disk with 512GB of storage.



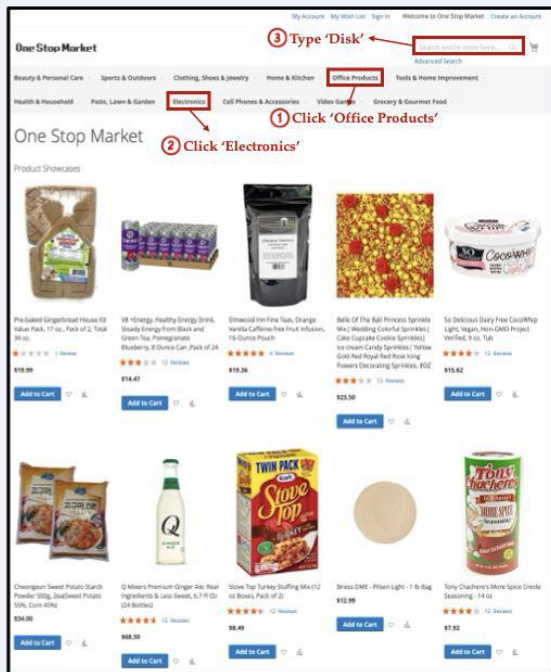
Stage I: Simulation

1

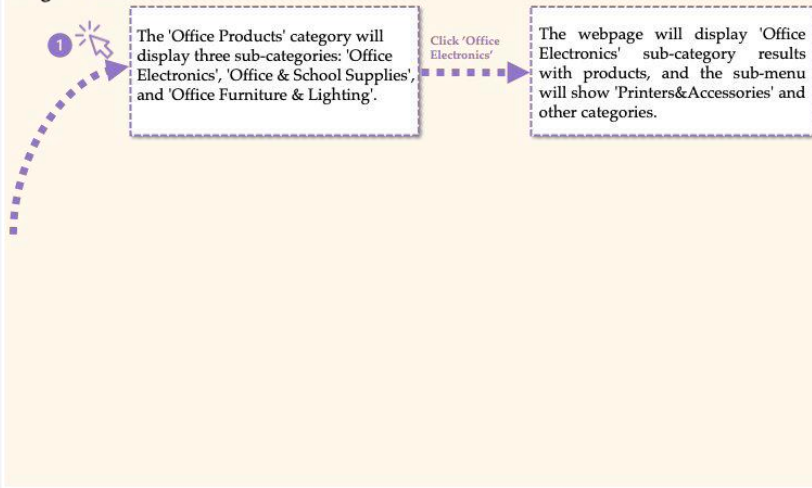
The 'Office Products' category will display three sub-categories: 'Office Electronics', 'Office & School Supplies', and 'Office Furniture & Lighting'.

WebDreamer: model-based planner

Please navigate to the 'Data Storage' category and purchase the least expensive disk with 512GB of storage.

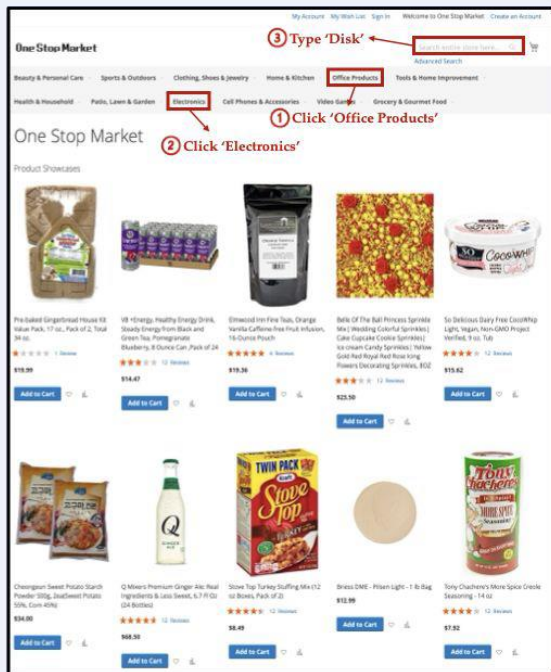


Stage I: Simulation

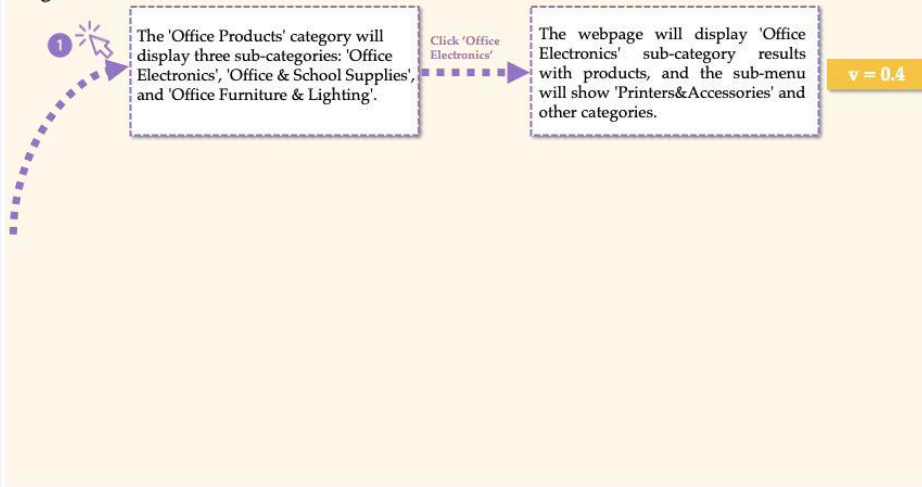


WebDreamer: model-based planner

Please navigate to the 'Data Storage' category and purchase the least expensive disk with 512GB of storage.

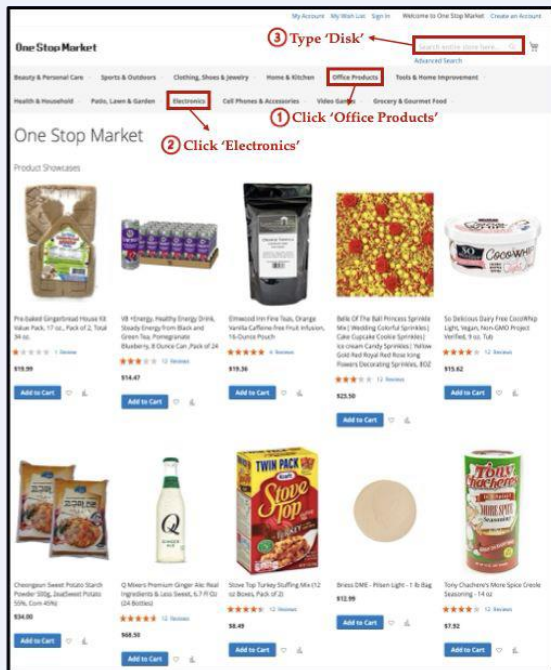


Stage I: Simulation

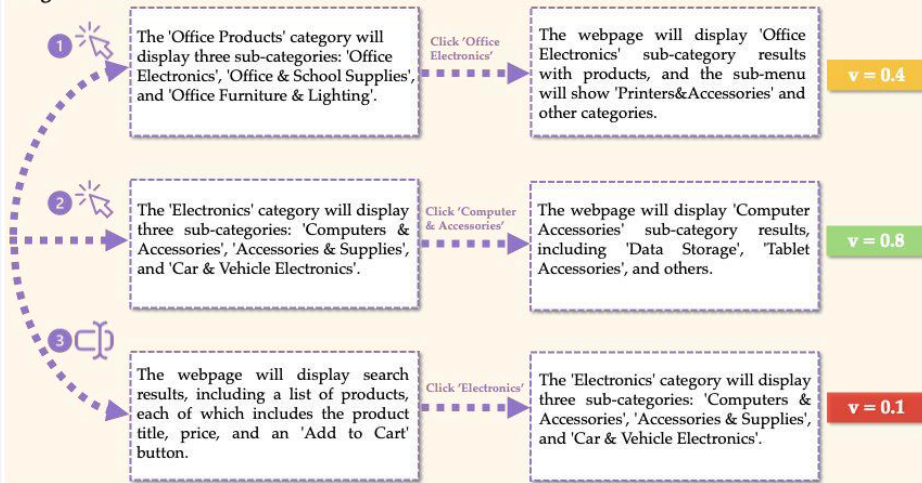


WebDreamer: model-based planner

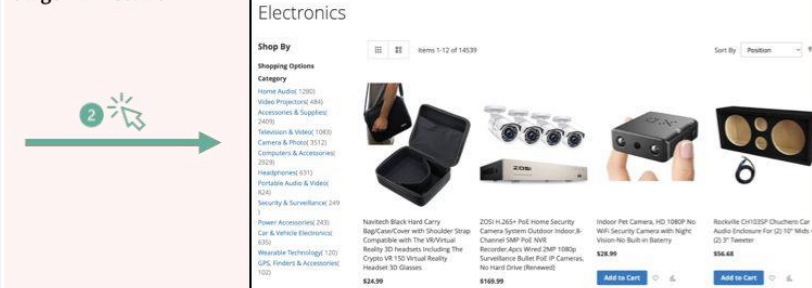
Please navigate to the 'Data Storage' category and purchase the least expensive disk with 512GB of storage.



Stage I: Simulation

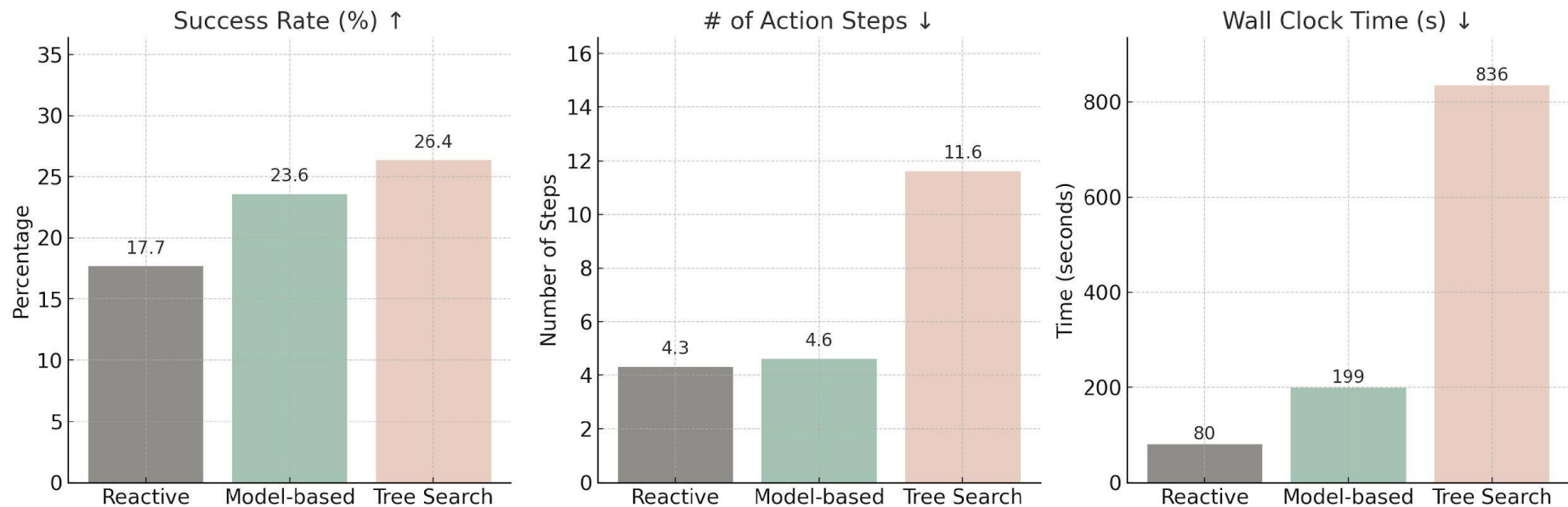


Stage II: Execution



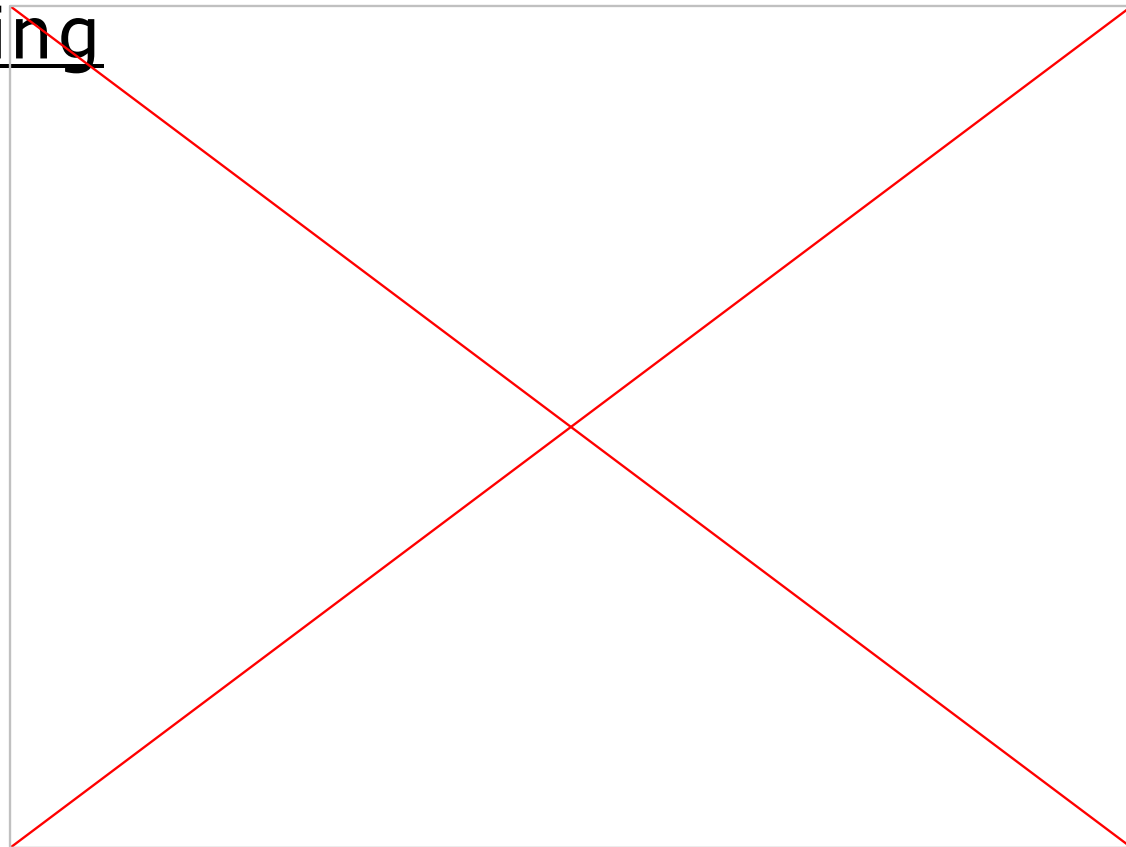
Results on VisualWebArena

Model-based planning is more accurate than reactive planning and more efficient than tree search (also recall the challenges of tree search in real-world environments)



Reactive and tree search results are from Koh et al., (2024)

Hierarchical planning and dynamic replanning



Hierarchical planning and dynamic rep

The screenshot displays the Devin AI interface within a web browser. The browser tab is titled 'Devin (the developer)' and the address bar shows 'preview.devin.ai'. The interface is divided into two main sections: 'Benchmark Llama 2 Providers' on the left and 'Devin's Workspace' on the right.

Benchmark Llama 2 Providers

Scott
Hey Devin, I'd like for you to benchmark Llama 2 on three different providers: Replicate, Together, and Perplexity. Figure out their API formats and write a script that sends the same prompt/params to all of them.

Devin
Got it, I'll look into the API formats for Replicate, Together, and Perplexity and create a script to benchmark Llama 2 across these services. I'll get started on that now.

Devin is researching the API documentation for Replicate, Together, and Perplexity to understand their request formats and parameters.

Send a message (won't interrupt Devin)

Devin's Workspace

Shell Browser Editor **Planner**

Current Focus: Research the API documentation for Replicate, Together, and Perplexity.

Plan:

- ☐ Research the API documentation for Replicate, Together, and Perplexity to understand their request formats and parameters.
- ☐ Write a Python script that constructs and sends HTTP requests to each API with the same prompt and parameters.
- ☐ Implement response time measurement and accuracy assessment within the script to benchmark each API's performance.
- ☐ Test the script to ensure it works correctly and gathers the necessary data from each API.
- ☐ Analyze the collected data to compare the performance of the three APIs.
- ☐ Document the script and the benchmarking process, including any dependencies or environment setup needed.
- ☐ Report the findings back to the user with the collected data and any relevant visualizations or summaries.

High-level Plan

Devin is in private beta and may make mistakes.

Hierarchical planning and dynamic rep

The screenshot displays the 'Devin's Workspace' interface. On the left, a chat window shows a conversation between 'Scott' and 'Devin'. Scott asks for confirmation on model names, and Devin responds, confirming the update of the Replicate model ID and mentioning a troubleshooting issue with the script. Below the chat, a URL bar shows 'https://docs.together.ai/reference/chat-completions'. The main area on the right contains a code editor. The 'default' tab shows a terminal output with a 'Command 'python' not found' error and a traceback from 'benchmark.py'. The 'benchmark.py' tab shows the source code of the script, which includes imports for 'os', 'time', and 'requests', and defines API endpoints for Replicate, Together AI, and Perplexity. A red box highlights the terminal error message.

Devin's Workspace

Scott
can you make sure you use the right model names? i want them to all be llama 2 70b.

Devin
Hey Scott, absolutely. I'll ensure that all the model names used in the benchmarking script are for Llama 2 70b. I've already updated the Replicate model ID with the one you provided. I'll double-check the model names for Together and Perplexity to make sure they're correct and update them if necessary. I'm currently troubleshooting an issue with the script and will proceed with the testing as soon as it's resolved. Thanks for the reminder!

Devin is currently debugging the script to handle API response errors and ensure accurate parsing of the output.

<https://docs.together.ai/reference/chat-completions>

terminal

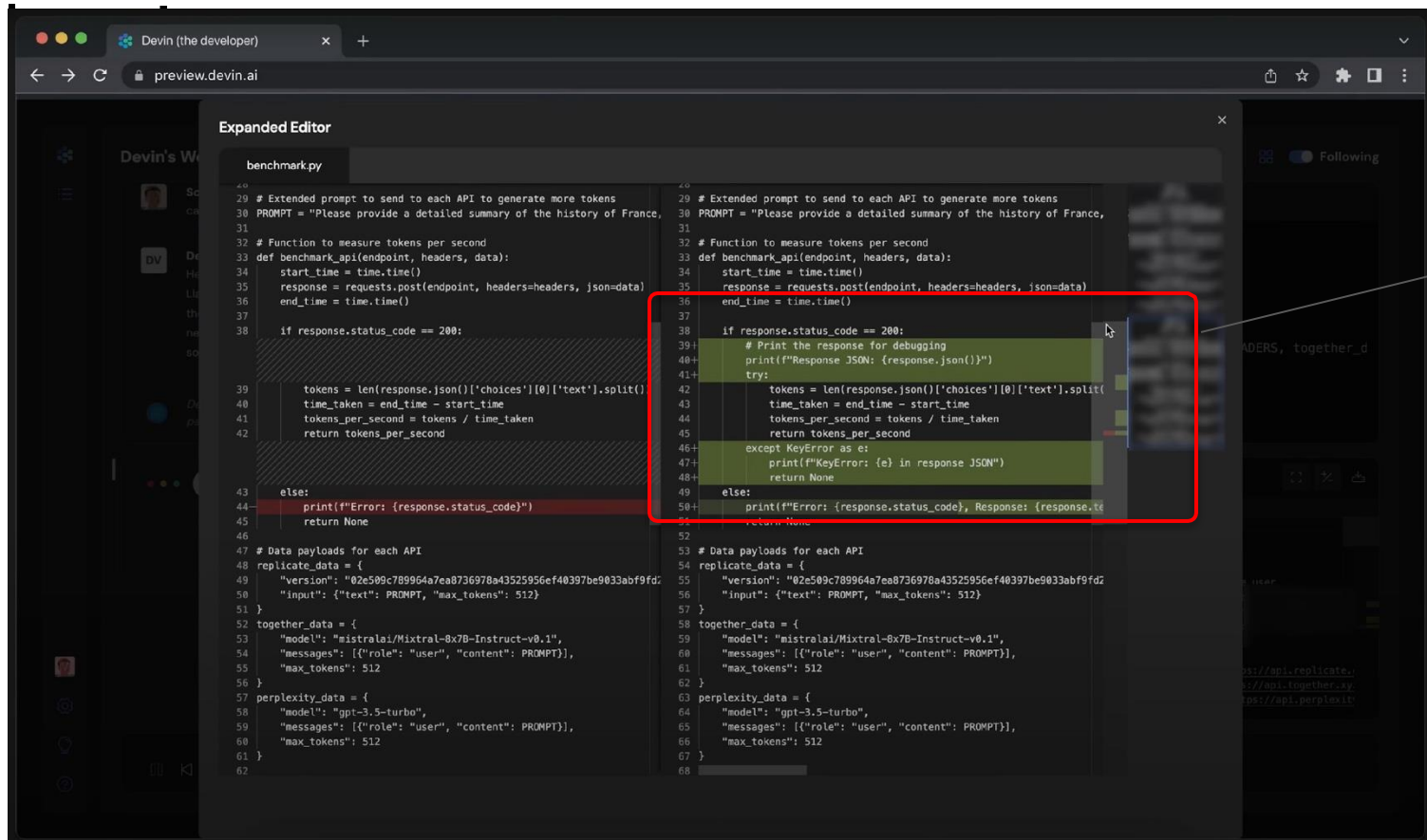
```
Command 'python' not found, did you mean:
command 'python3' from deb python3
command 'python' from deb python-is-python3
ubuntu@ip-172-31-27-196:~$ python3 benchmark.py
Error: 422
Traceback (most recent call last):
  File "/home/ubuntu/benchmark.py", line 65, in <module>
    together_tps = benchmark_api(TOGETHER_ENDPOINT, TOGETHER_HEADERS, together_d
ata)
  File "/home/ubuntu/benchmark.py", line 39, in benchmark_api
    tokens = len(response.json()['choices'][0]['text'].split())
KeyError: 'text'
ubuntu@ip-172-31-27-196:~$
```

benchmark.py

```
home > ubuntu > benchmark.py
1 import os
2 import time
3 import requests
4
5 # API keys provided by the user
6 TOGETHER_API_KEY = "c"
7 REPLICATE_API_KEY = "r"
8 PERPLEXITY_API_KEY = "p"
9
10 # API endpoints
11 REPLICATE_ENDPOINT = "https://api.replicate.
12 TOGETHER_ENDPOINT = "https://api.together.xy
13 PERPLEXITY_ENDPOINT = "https://api.perplexit
14
```

Ran into exception when carrying out a subgoal

Hierarchical planning and dynamic rep



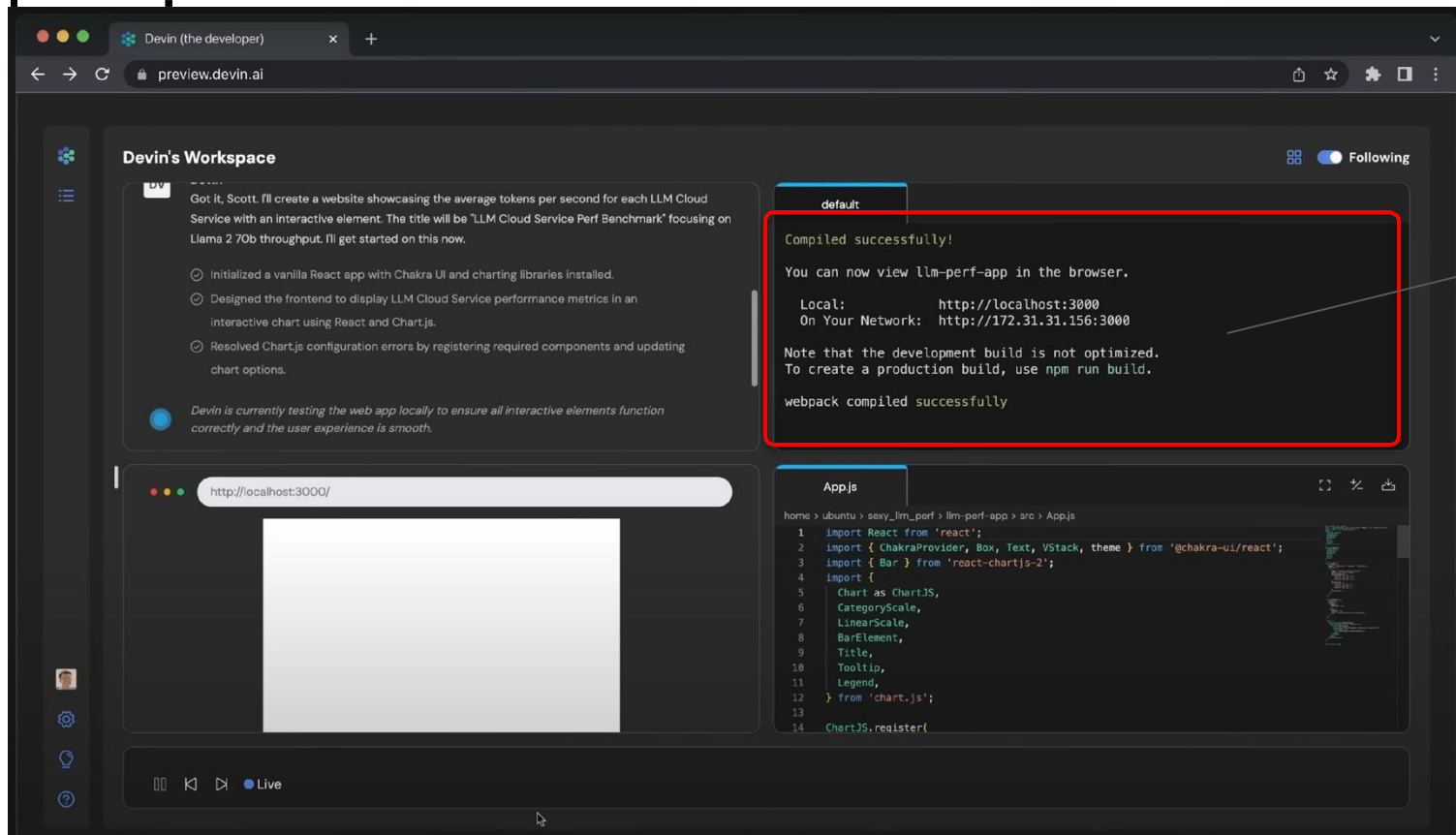
The screenshot shows the 'Expanded Editor' in the Devin AI interface. It displays two versions of a Python script named 'benchmark.py'. The left version shows the original code, and the right version shows a modified version. A red box highlights a change in the exception handling logic. An arrow points from the text 'Re-plan and add print() around the exception to get debug info' to the highlighted code.

```
29 # Extended prompt to send to each API to generate more tokens
30 PROMPT = "Please provide a detailed summary of the history of France,
31
32 # Function to measure tokens per second
33 def benchmark_api(endpoint, headers, data):
34     start_time = time.time()
35     response = requests.post(endpoint, headers=headers, json=data)
36     end_time = time.time()
37
38     if response.status_code == 200:
39
40         tokens = len(response.json()['choices'][0]['text'].split())
41         time_taken = end_time - start_time
42         tokens_per_second = tokens / time_taken
43         return tokens_per_second
44
45     else:
46         print(f"Error: {response.status_code}")
47         return None
48
49 # Data payloads for each API
50 replicate_data = {
51     "version": "02e509c789964a7ea8736978a43525956ef40397be9033ab9fd",
52     "input": {"text": PROMPT, "max_tokens": 512}
53 }
54 together_data = {
55     "model": "mistralai/Mistral-8x7B-Instruct-v0.1",
56     "messages": [{"role": "user", "content": PROMPT}],
57     "max_tokens": 512
58 }
59 perplexity_data = {
60     "model": "gpt-3.5-turbo",
61     "messages": [{"role": "user", "content": PROMPT}],
62     "max_tokens": 512
63 }
```

```
29 # Extended prompt to send to each API to generate more tokens
30 PROMPT = "Please provide a detailed summary of the history of France,
31
32 # Function to measure tokens per second
33 def benchmark_api(endpoint, headers, data):
34     start_time = time.time()
35     response = requests.post(endpoint, headers=headers, json=data)
36     end_time = time.time()
37
38     if response.status_code == 200:
39         # Print the response for debugging
40         print(f"Response JSON: {response.json()}")
41         try:
42             tokens = len(response.json()['choices'][0]['text'].split())
43             time_taken = end_time - start_time
44             tokens_per_second = tokens / time_taken
45             return tokens_per_second
46         except KeyError as e:
47             print(f"KeyError: {e} in response JSON")
48             return None
49     else:
50         print(f"Error: {response.status_code}, Response: {response.json()}")
51         return None
52
53 # Data payloads for each API
54 replicate_data = {
55     "version": "02e509c789964a7ea8736978a43525956ef40397be9033ab9fd",
56     "input": {"text": PROMPT, "max_tokens": 512}
57 }
58 together_data = {
59     "model": "mistralai/Mistral-8x7B-Instruct-v0.1",
60     "messages": [{"role": "user", "content": PROMPT}],
61     "max_tokens": 512
62 }
63 perplexity_data = {
64     "model": "gpt-3.5-turbo",
65     "messages": [{"role": "user", "content": PROMPT}],
66     "max_tokens": 512
67 }
```

Re-plan and add print() around the exception to get debug info

Hierarchical planning and dynamic rep



Solved the exception based on the printed out debug info!

Planning: Takeaways

- Language agents are expanding into new planning scenarios
 - characterized by expressive but fuzzy goal specifications, open-ended action spaces, more difficult and sometimes non-binary goal tests
- Language for reasoning also enables new planning abilities
 - Generalist world models and model-based planning
 - Hierarchical planning and dynamic replanning
- The best planning strategy is dependent on the LLM; stronger LLMs may require less scaffolding (i.e., more 'reactive')
- How to improve planning in LLMs is still largely an open question