Multimodal Autonomous AI Agents

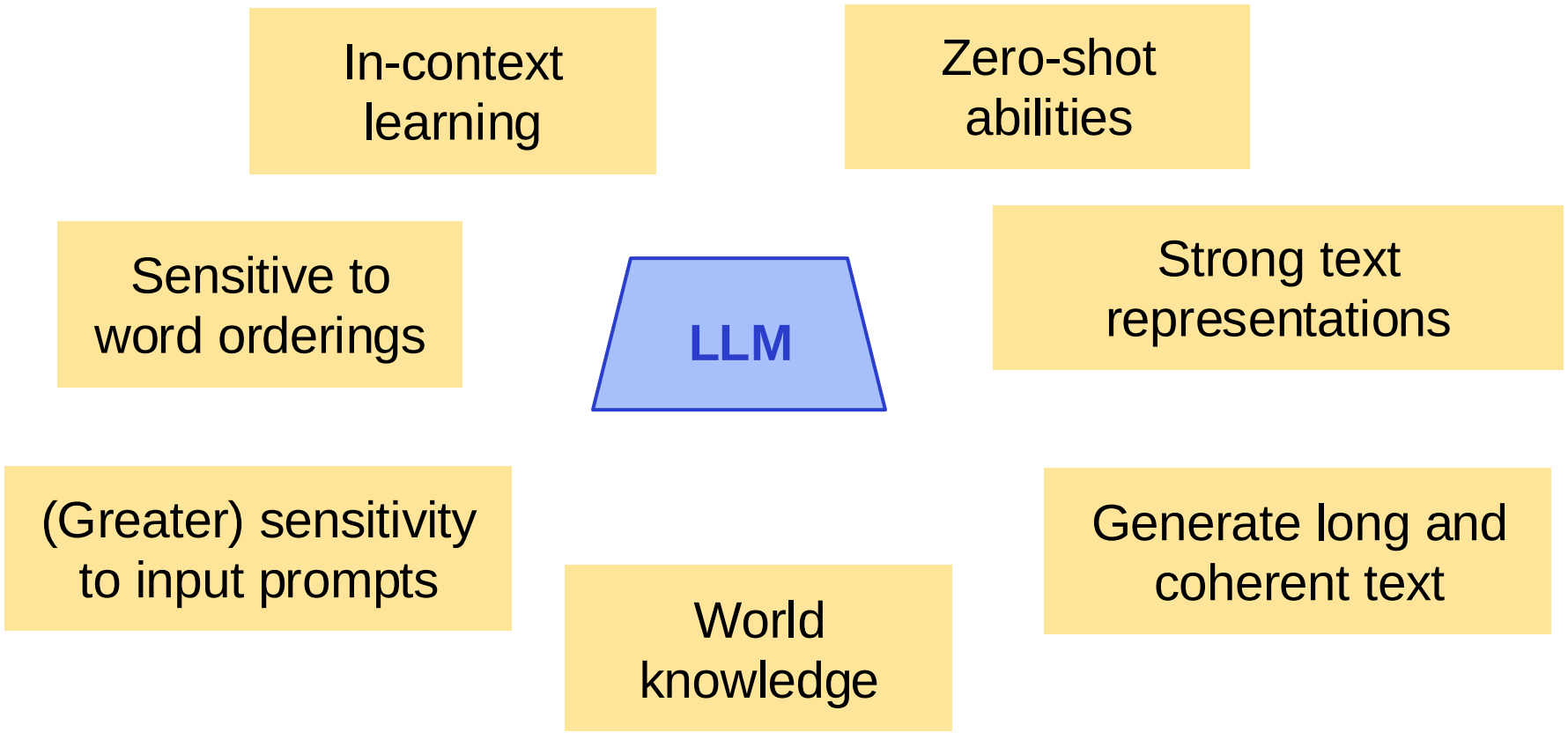
Russ Salakhutdinov

Machine Learning Department
Carnegie Mellon University

Carnegie Mellon University

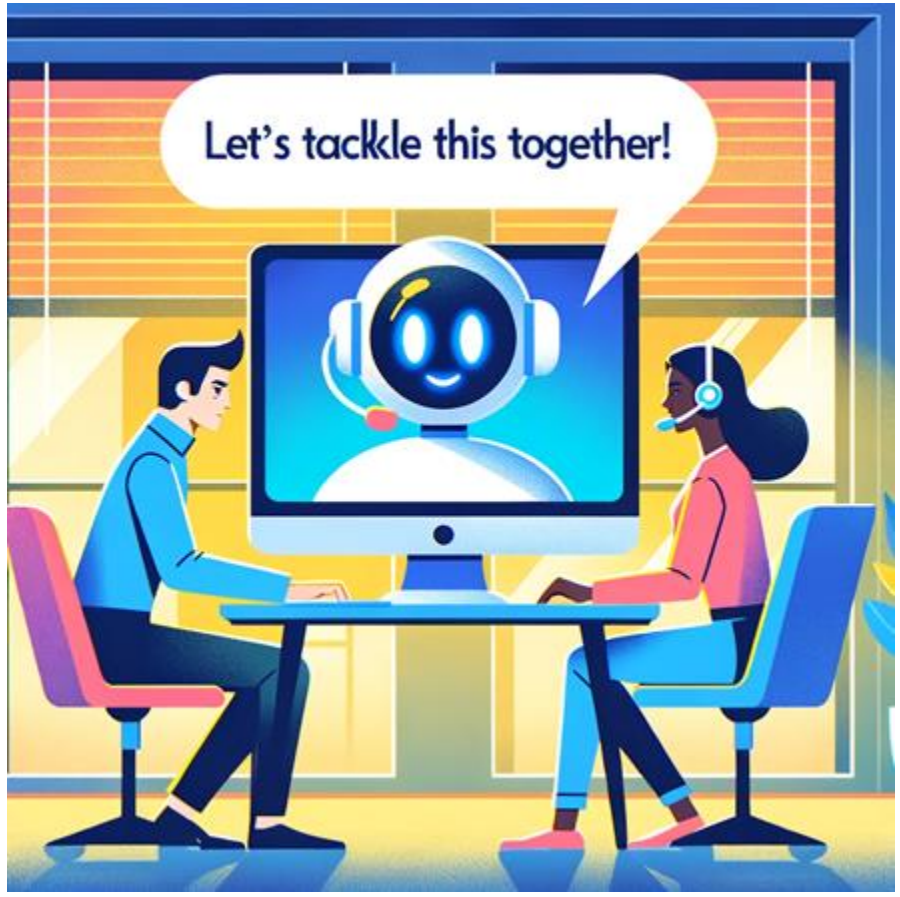
MACHINE LEARNING DEPARTMENT

Large Language Models



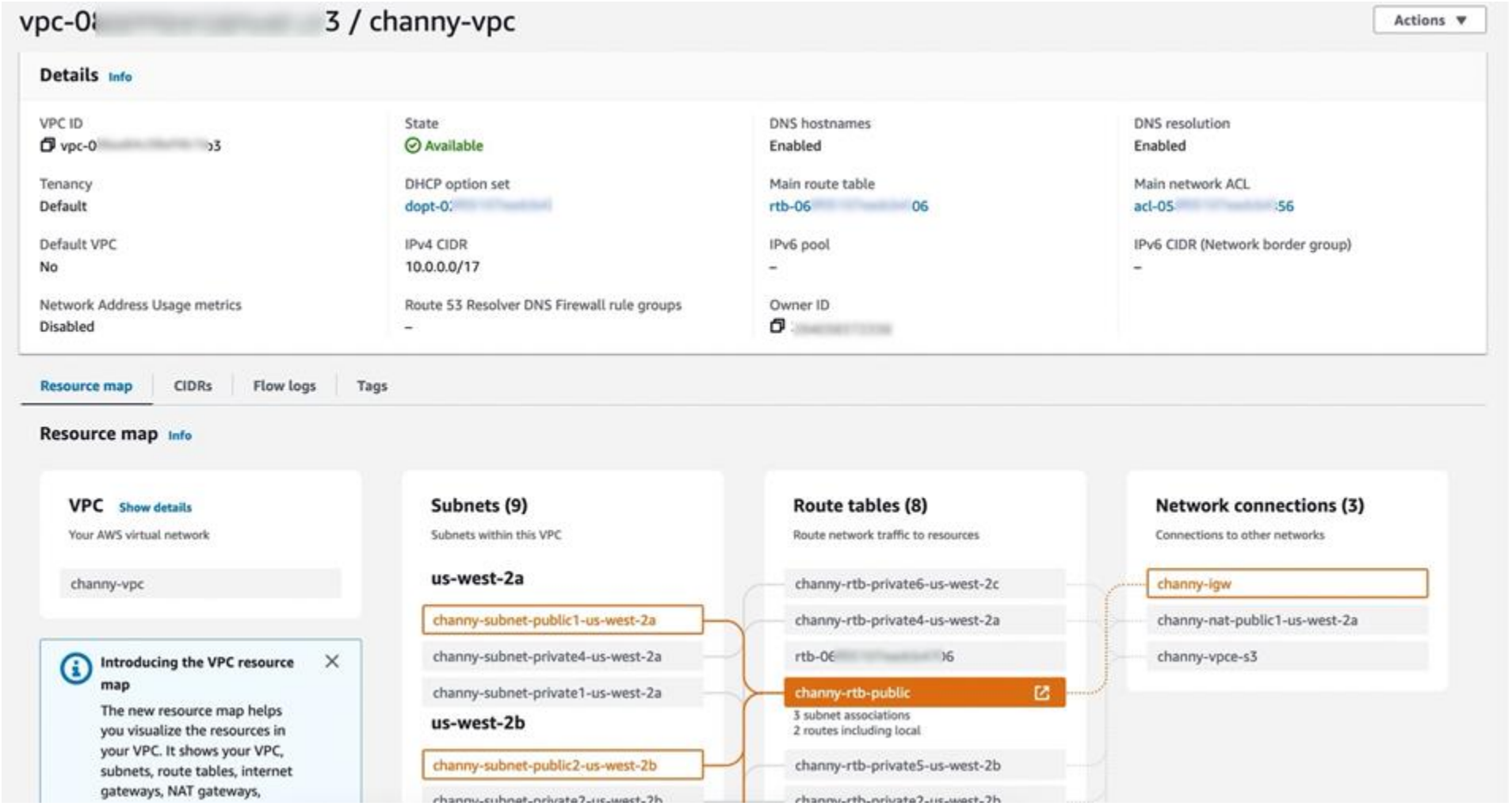
Autonomous AI Agents

- Many productive tasks we perform today are done on the computer
  - And many of these are on the web
- Many opportunities to automate menial tasks
- Augment human capabilities

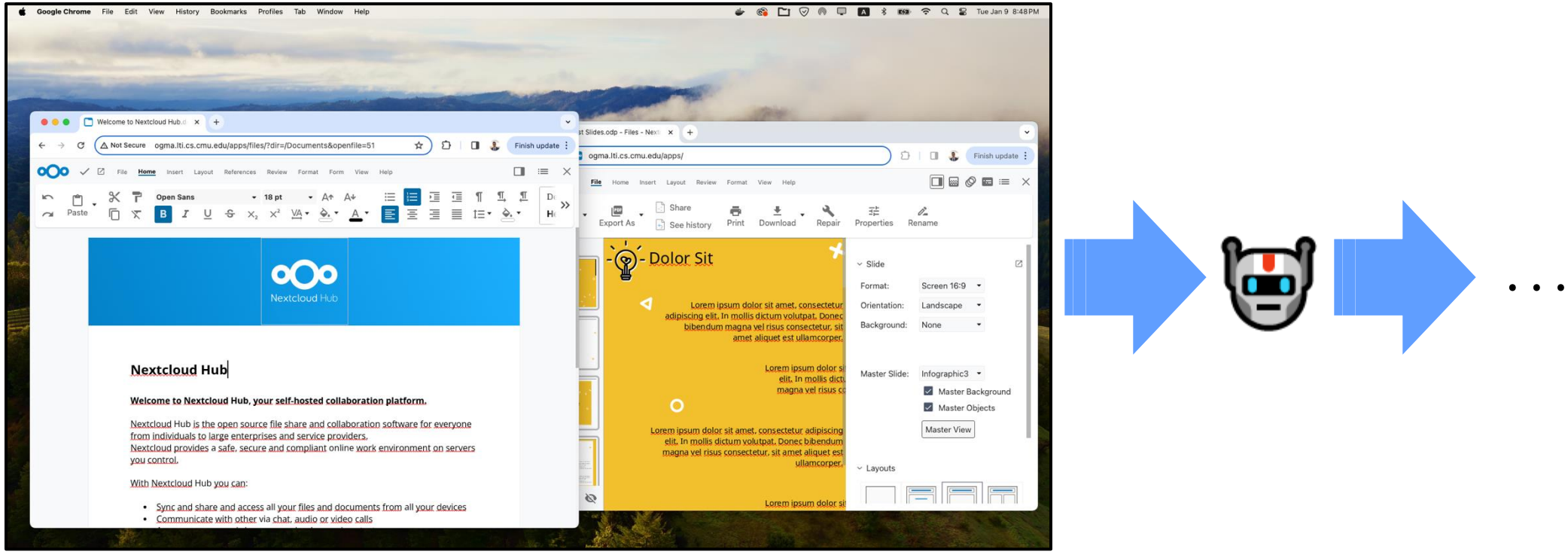


Generated with DALLE

Autonomous Agents

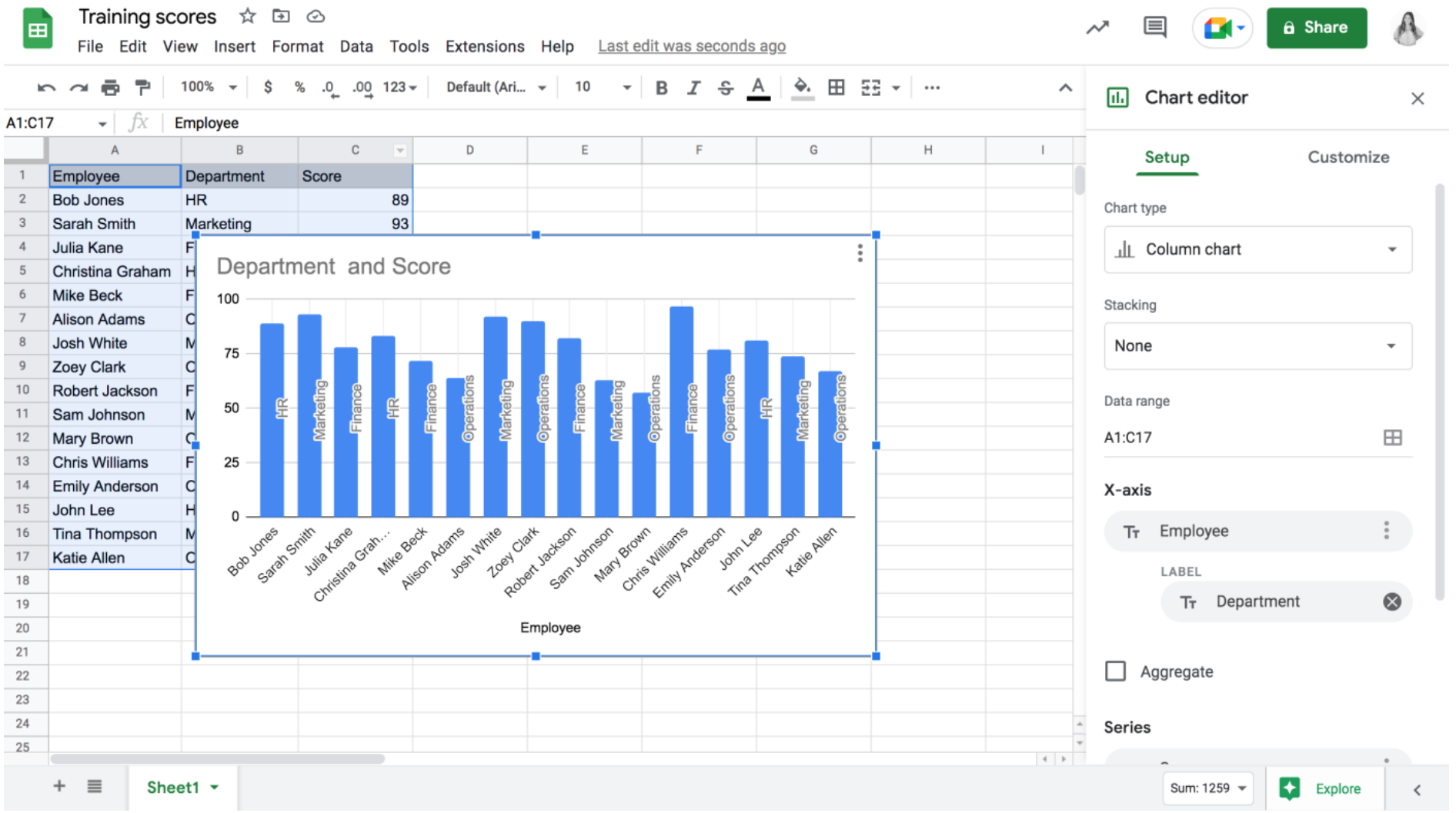


Autonomous Agents



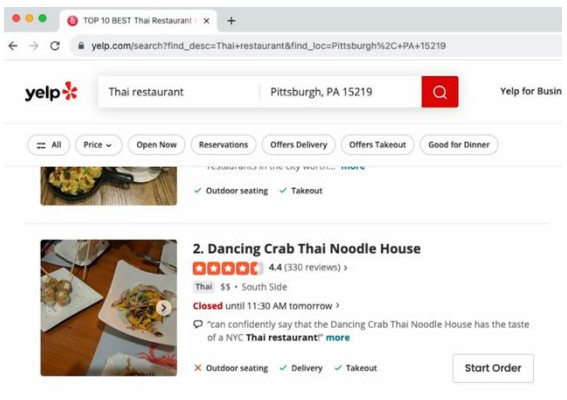
Task: “Create a set of PowerPoint slides to present the content in this paper.”

Autonomous Agents



Web Agents

Web Grounding

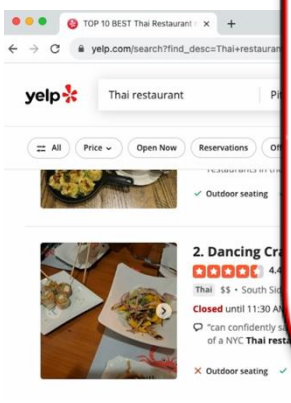


Visual Encoder + LLM

HTML Understanding

Web Agents

Web



Shunyu Yao, REACT Synergizing Reasoning and Acting in Language Models, 2023

Jason Wei et al, Chain of Thought Prompting Elicits Reasoning in Large Language Models, 2022

Reiichiro Nakano et al, WebGPT: Browser-assisted Question-Answering with Human Feedback, 2021.

Xiang Deng et al, MIND2WEB: Towards a Generalist Agent for the Web, 2023

Timo Schick et al, Toolformer: Language Models can Teach Themselves to Use Tools, 2023

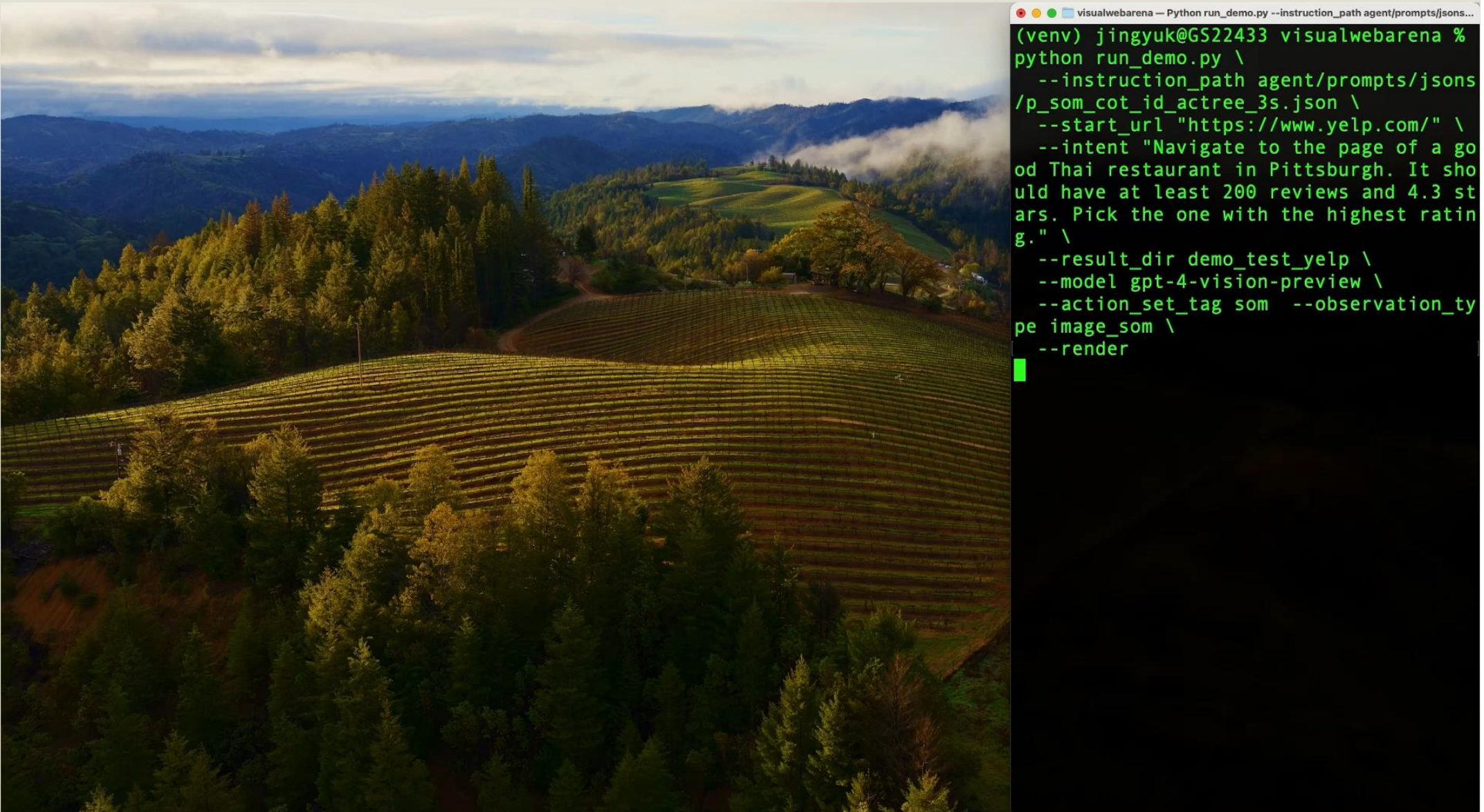
Shibo Hao et al, ToolkenGPT: Augmenting Frozen Language Models with Massive Tools via Tool Embeddings, 2023

Yang et al., SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering, 2024

Task: Navigate to a page of a good Thai restaurant in Pittsburgh. It should have at least 200 reviews and 4.3 stars. Pick the one with the highest rating

ellon University

Task: Navigate to the page of a good Thai restaurant in Pittsburgh. It should have at least 200 reviews and 4.3 stars. Pick the one with the highest rating.

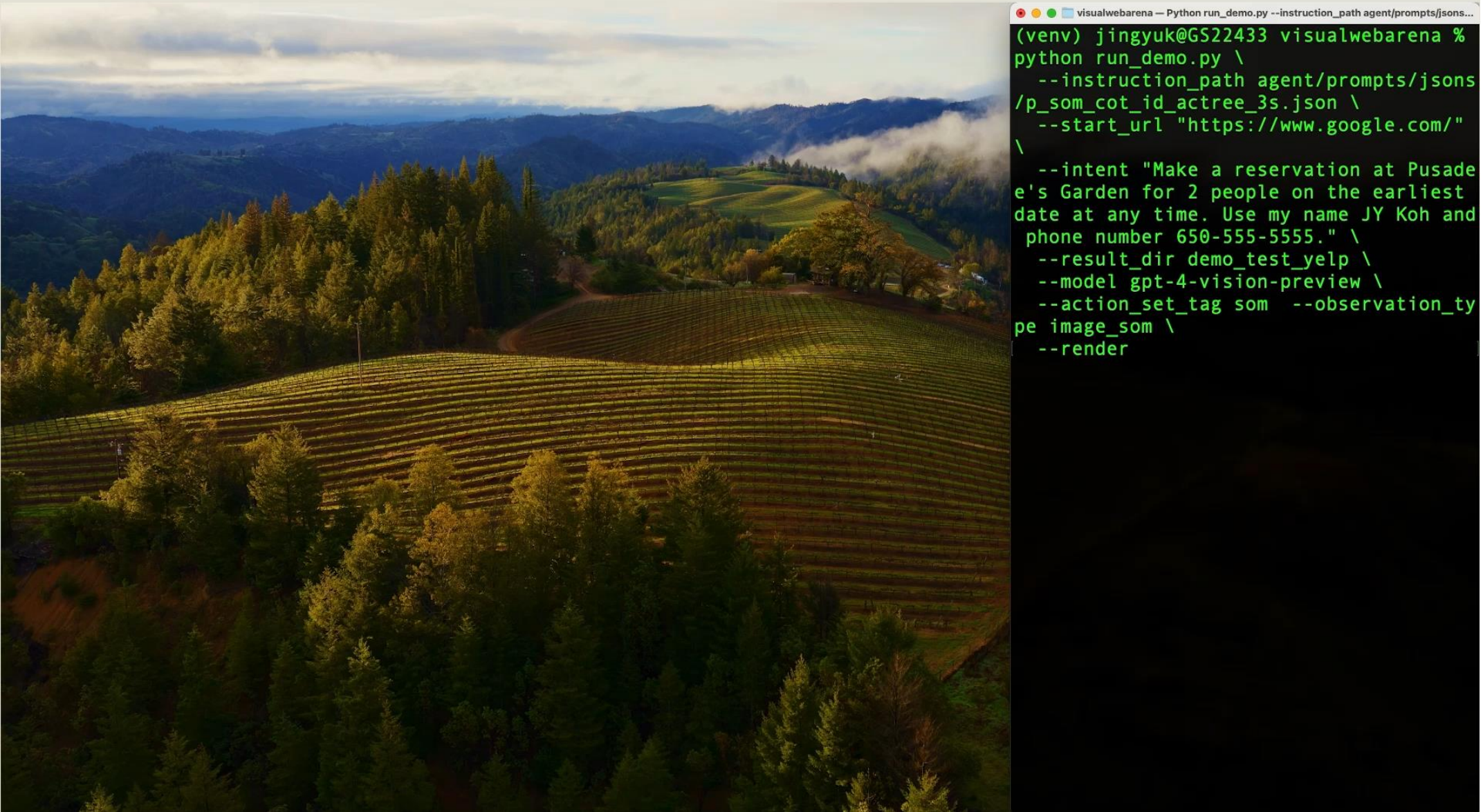


visualwebarena — Python run_demo.py --instruction_path agent/prompts/jsons...

```
(venv) jingyuk@GS22433 visualwebarena %
python run_demo.py \
  --instruction_path agent/prompts/jsons
/p_som_cot_id_actree_3s.json \
  --start_url "https://www.yelp.com/" \
  --intent "Navigate to the page of a go
od Thai restaurant in Pittsburgh. It sho
uld have at least 200 reviews and 4.3 st
ars. Pick the one with the highest ratin
g." \
  --result_dir demo_test_yelp \
  --model gpt-4-vision-preview \
  --action_set_tag som  --observation_ty
pe image_som \
  --render
```

ellon University

Task: Make a reservation at Pusadee's Garden for 2 people on the earliest date for dinner. Use my name JY Koh and phone number 650-555-5555.



visualwebarena — Python run_demo.py --instruction_path agent/prompts/jsons...

```
(venv) jingyuk@GS22433 visualwebarena %
python run_demo.py \
  --instruction_path agent/prompts/jsons
/p_som_cot_id_actree_3s.json \
  --start_url "https://www.google.com/"
\
  --intent "Make a reservation at Pusade
e's Garden for 2 people on the earliest
date at any time. Use my name JY Koh and
 phone number 650-555-5555." \
  --result_dir demo_test_yelp \
  --model gpt-4-vision-preview \
  --action_set_tag som  --observation_ty
pe image_som \
  --render
```

Task: Help me navigate to a shirt that has this on it.

ellon University

visualwebarena — Python run_demo.py --instruction_path agent/prompts/jsons...

```
(venv) jingyuk@GS22433 visualwebarena %
python run_demo.py \
  --instruction_path agent/prompts/jsons
/p_som_cot_id_actree_3s.json \
  --start_url "https://www.amazon.com/"
\
  --image "https://media.npr.org/assets/
img/2023/01/14/this-is-fine_wide-0077dc0
607062e15b476fb7f3bd99c5f340af356-s1400-
c100.jpg" \
  --intent "Help me navigate to a shirt
that has this on it." \
  --result_dir demo_test_amazon \
  --model gpt-4-vision-preview \
  --action_set_tag som  --observation_ty
pe image_som \
  --render
```

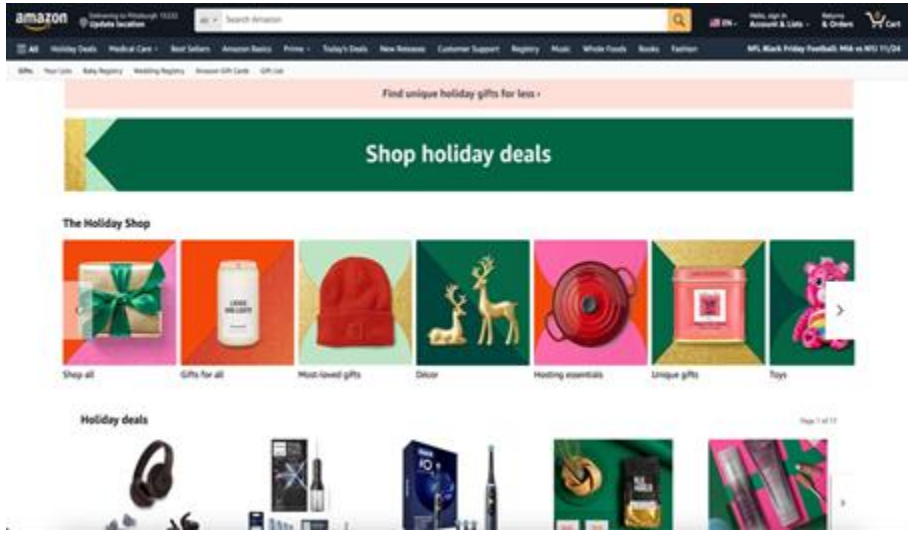
WebArena

Shuyan Zhou Frank Xu

- Most realistic web environment at the moment
- Websites from popular categories (shopping, Reddit, GitLab)
  - Self-hosted open source re-implementations
  - Data from real websites (Amazon, Reddit, GitHub)
- Tasks are easy for humans (78% success rate) but difficult for language model agents (14%)
- **But:** Tasks are designed to use just text and HTML source code
- Messy HTML, JavaScript: usually minified or compressed for efficiency
- Interactive elements don't display correctly in HTML
  - e.g., JavaScript/CSS code that moves objects after the page is loaded
- Context length: HTML pages are complex, easily filling up > 100k tokens

Zhou*, Xu*, et al., 2023. "WebArena: A Realistic Web Environment for Building Autonomous Agents"

HTML is insufficient



- Messy HTML, JavaScript: usually minified or compressed for efficiency
- Interactive elements don't display correctly in HTML
  - e.g., JavaScript/CSS code that moves objects after the page is loaded
  - Spatial layout is also usually not conveyed well
- Context length: HTML pages are complex, easily filling up > 100k tokens

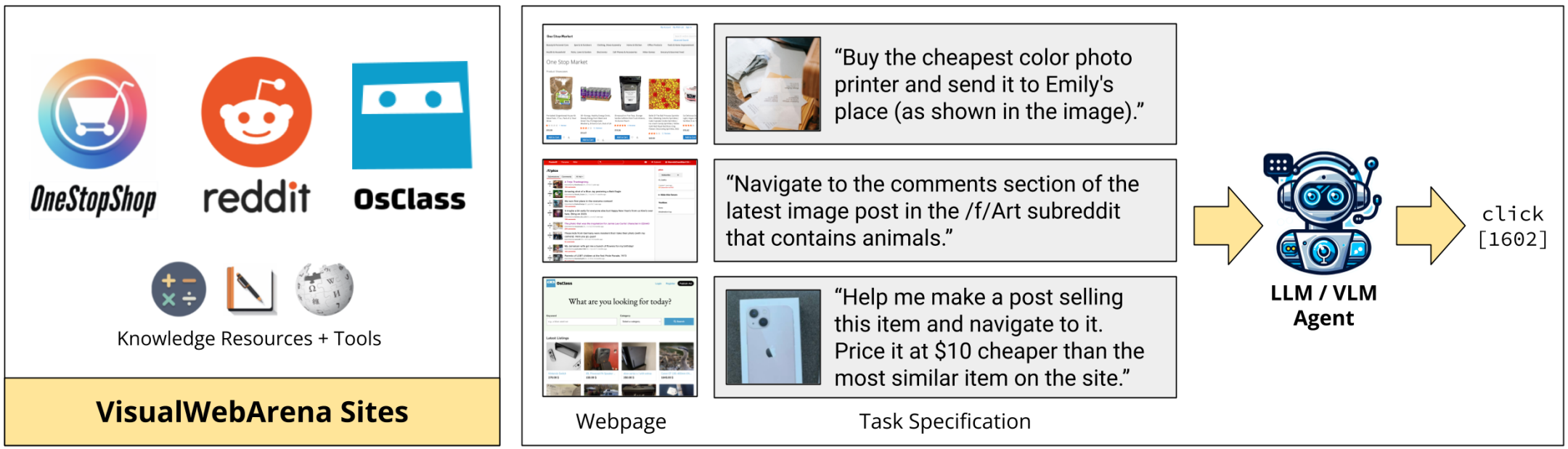
VisualWebArena


Jing Yu Koh


Shuyan Zhou


Frank Xu

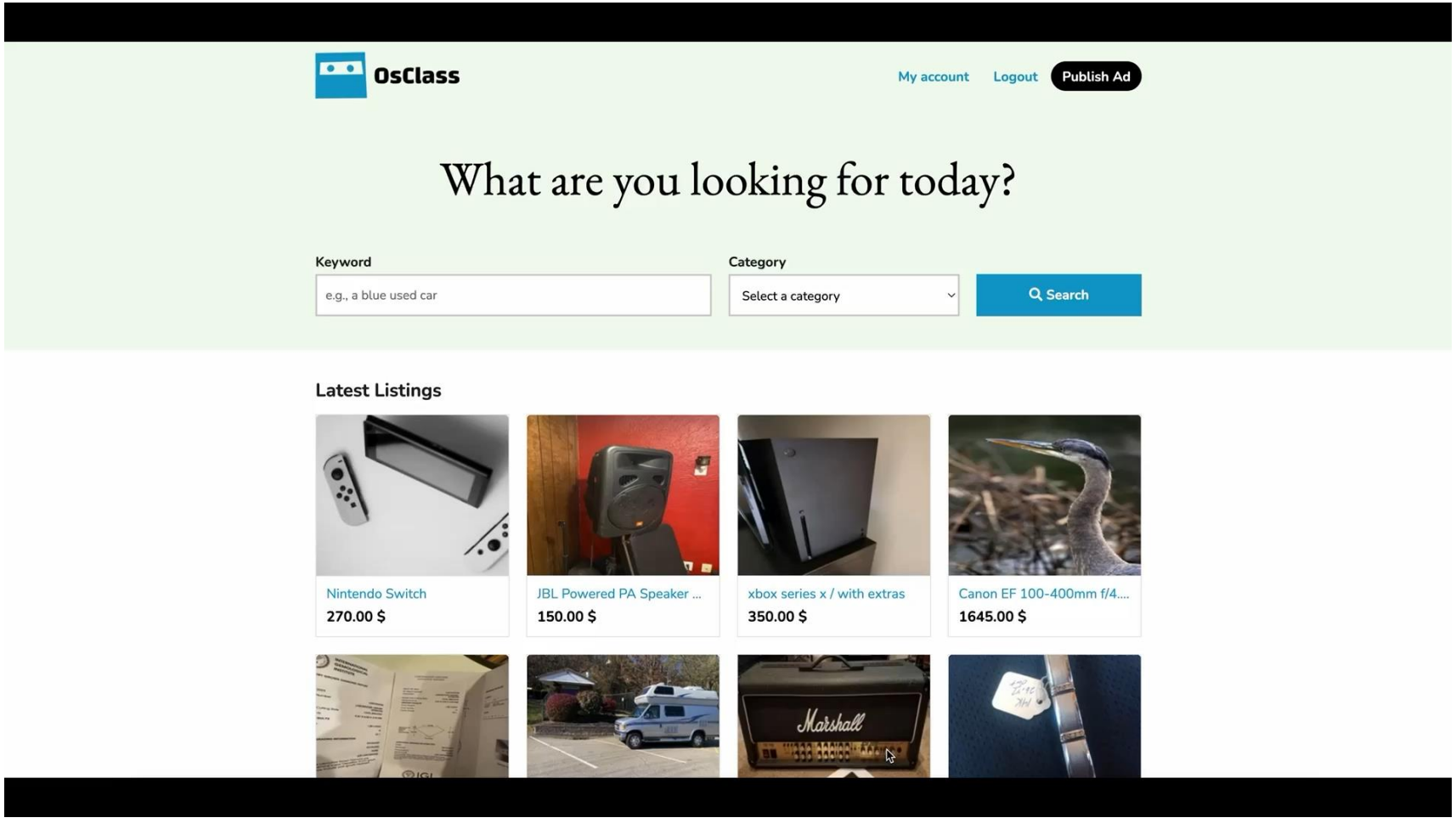
- Build and track the progress of **multimodal agents**
- We design visually grounded tasks to test these abilities
- Visual inputs (and outputs) allow for unique, interesting, and realistic tasks



JY Koh, V. Duvvur, P. Huang, L. Jang, MC Lim, R. Lo, G. Neubig, S. Zhou, R. Salakhutdinov, D. Fried

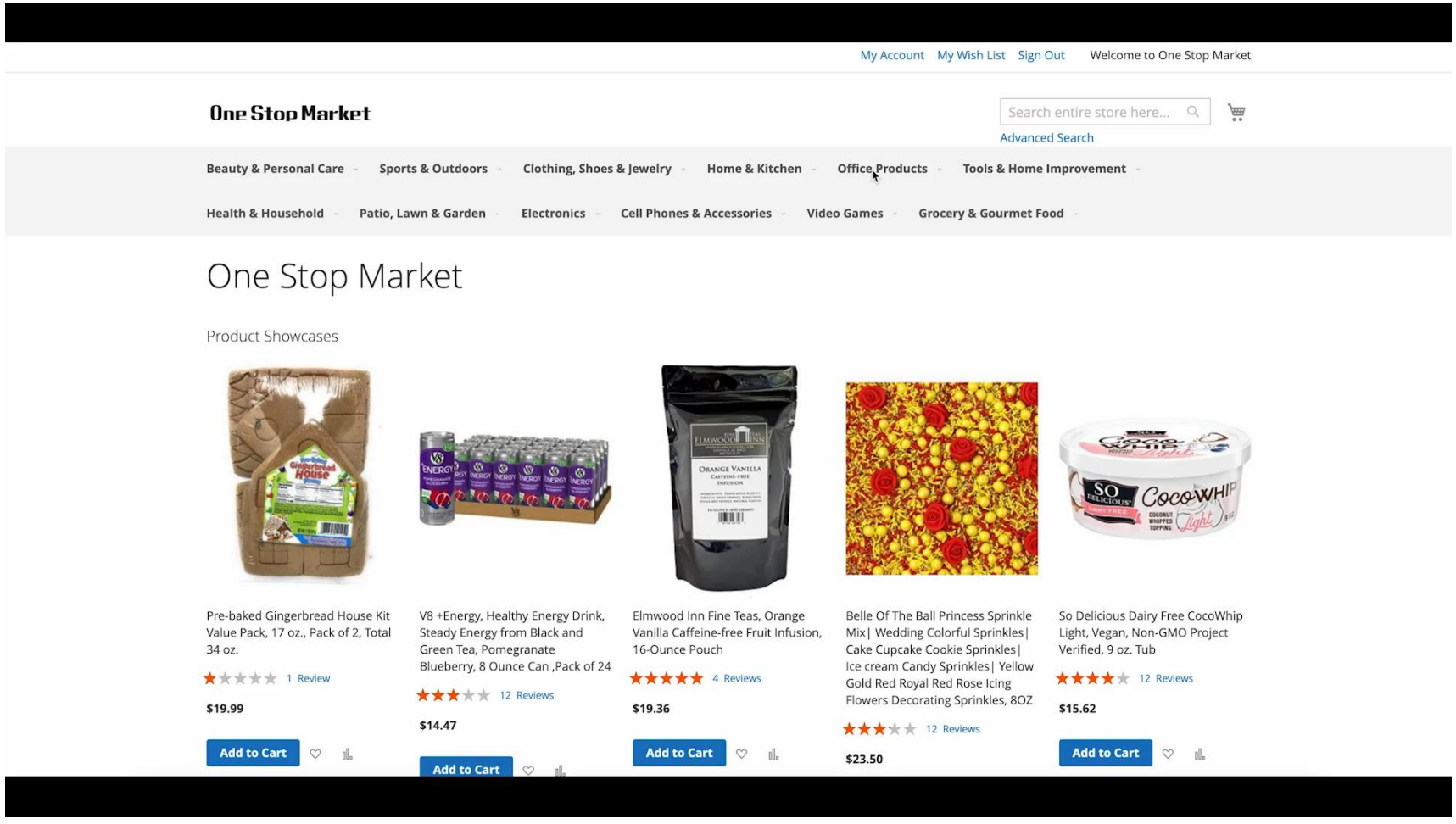
VisualWebArena: Classifields

Task: Find this exact bike that's listed for $300-500 and post a comment offering $10 less than their asking price.

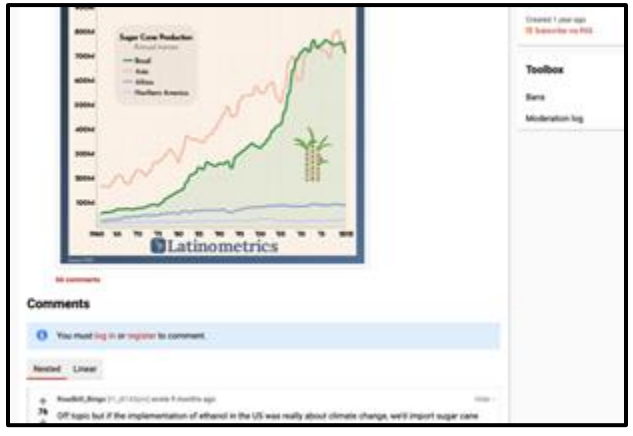


VisualWebArena: Shopping

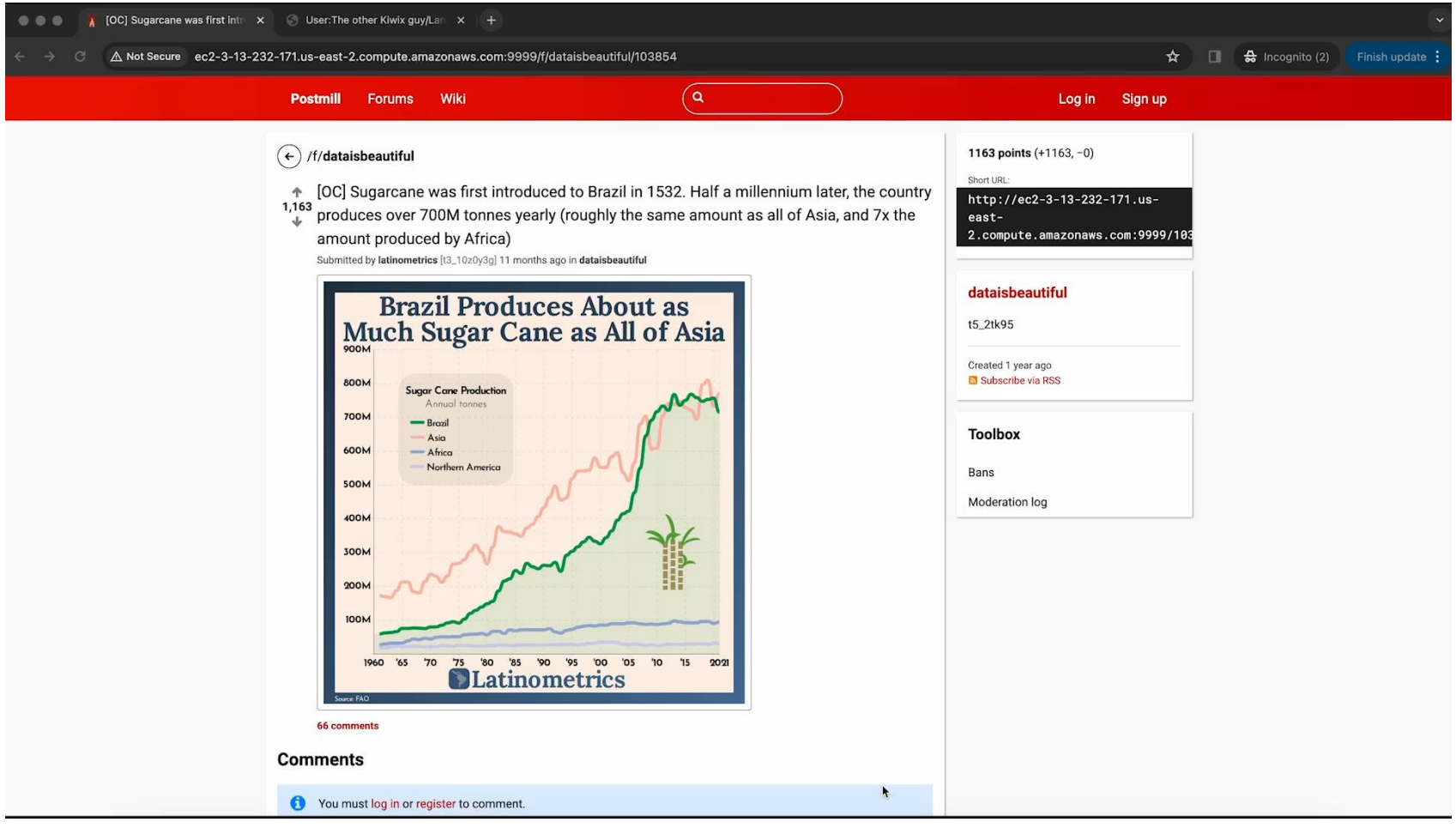
Task: Buy the cheapest color photo printer and send it to Emily's place (as shown in the image).



VisualWebArena: Reddit



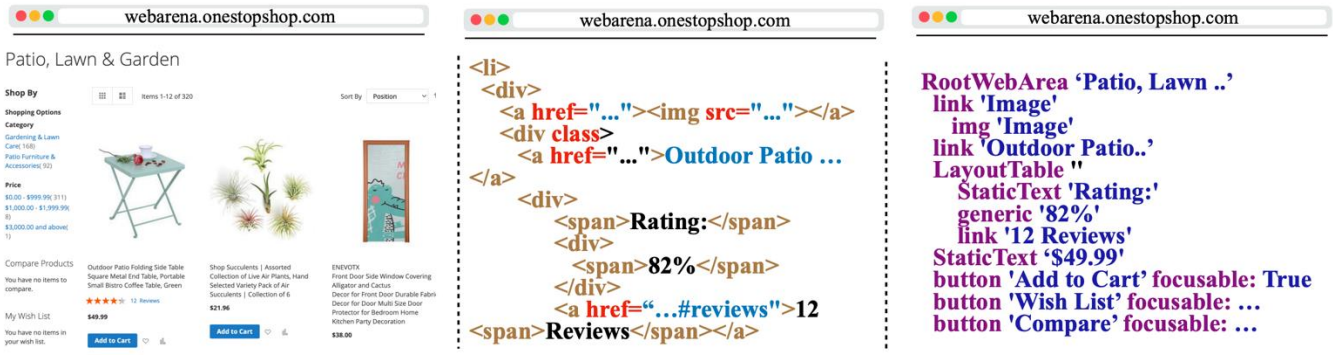
Task: What is the 2022 total nominal GDP of the area that produces most sugarcane in the year of 2021? (in billion)?



VisualWebArena

POMDP environment: $\mathcal{E} = \langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T} \rangle$

- Observations $\mathcal{O}$



- Actions $\mathcal{A}$

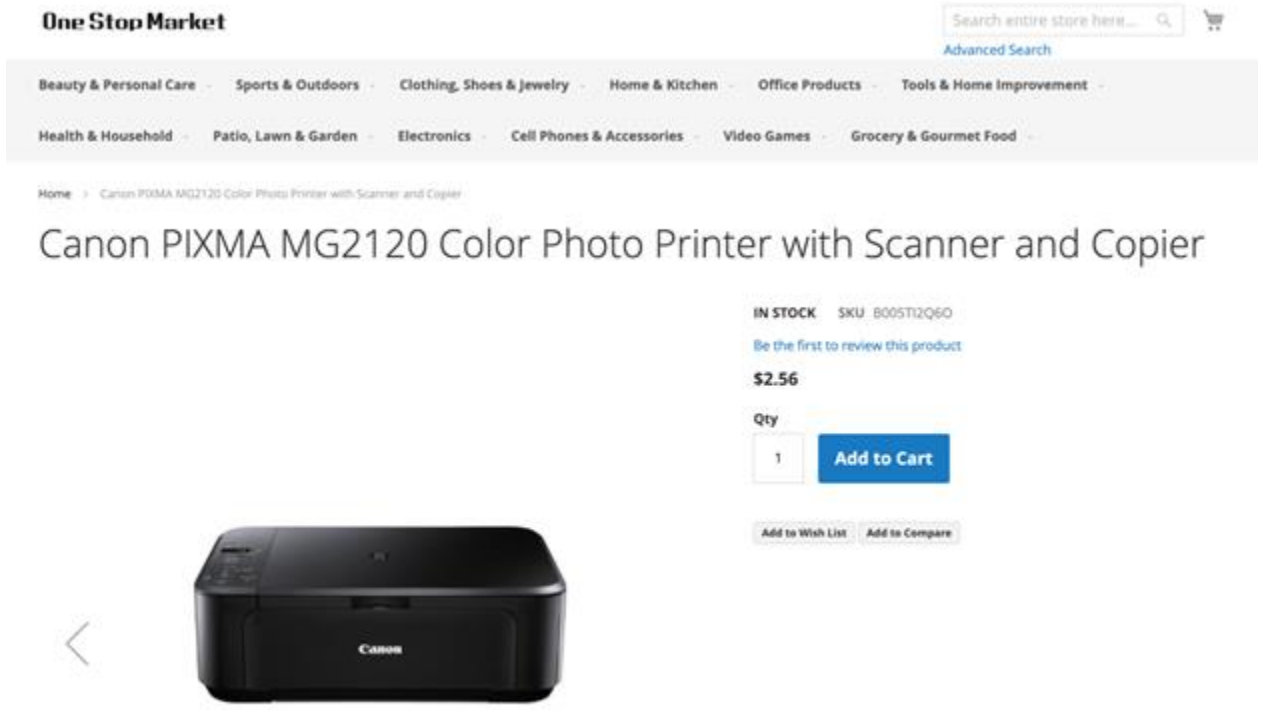
Action Type a	Description
`click [elem]`	Click on element `elem`.
`hover [elem]`	Hover on element `elem`.
`type [elem] [text]`	Type `text` on element `elem`.
`press [key_comb]`	Press a key combination.
`new_tab`	Open a new tab.
`tab_focus [index]`	Focus on the i-th tab.
`tab_close`	Close current tab.
`goto [url]`	Open `url`.
`go_back`	Click the back button.
`go_forward`	Click the forward button.
`scroll [up|down]`	Scroll up or down the page.
`stop [answer]`	End the task with an optional output.

- Deterministic transition function

$$\mathcal{T} : \mathcal{S} \times \mathcal{A} \longrightarrow \mathcal{S}$$

- Reward function: $r(\mathbf{a}, \mathbf{s})$

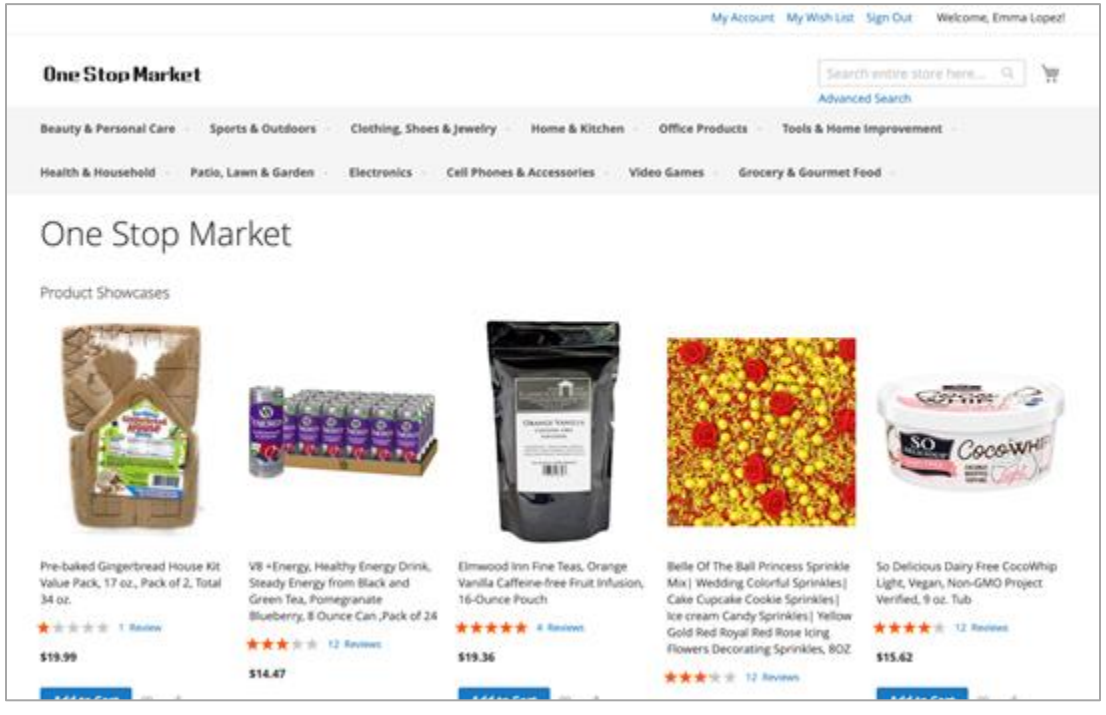
Image Inputs:



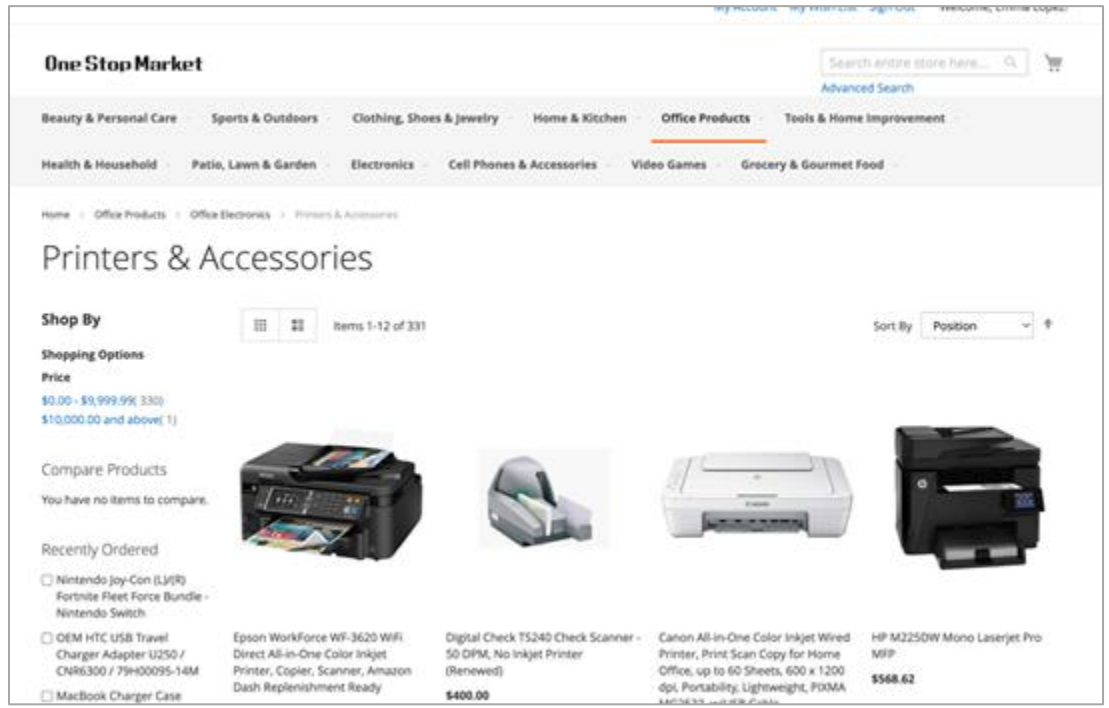
Task: "Buy the cheapest color photo printer and send it to Emily's place (as shown in the image)."

Shopping

Task: “Buy the cheapest color photo printer and send it to Emily's place (as shown in the image).”

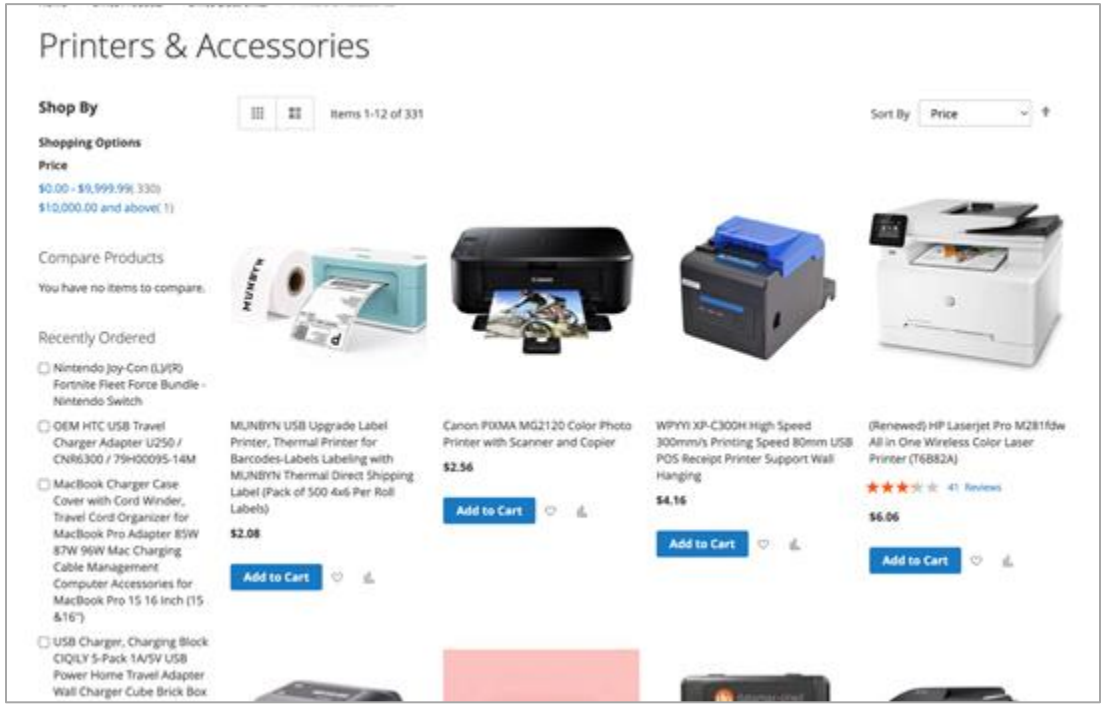


Step 0: Start on the homepage of OneStopMarket.

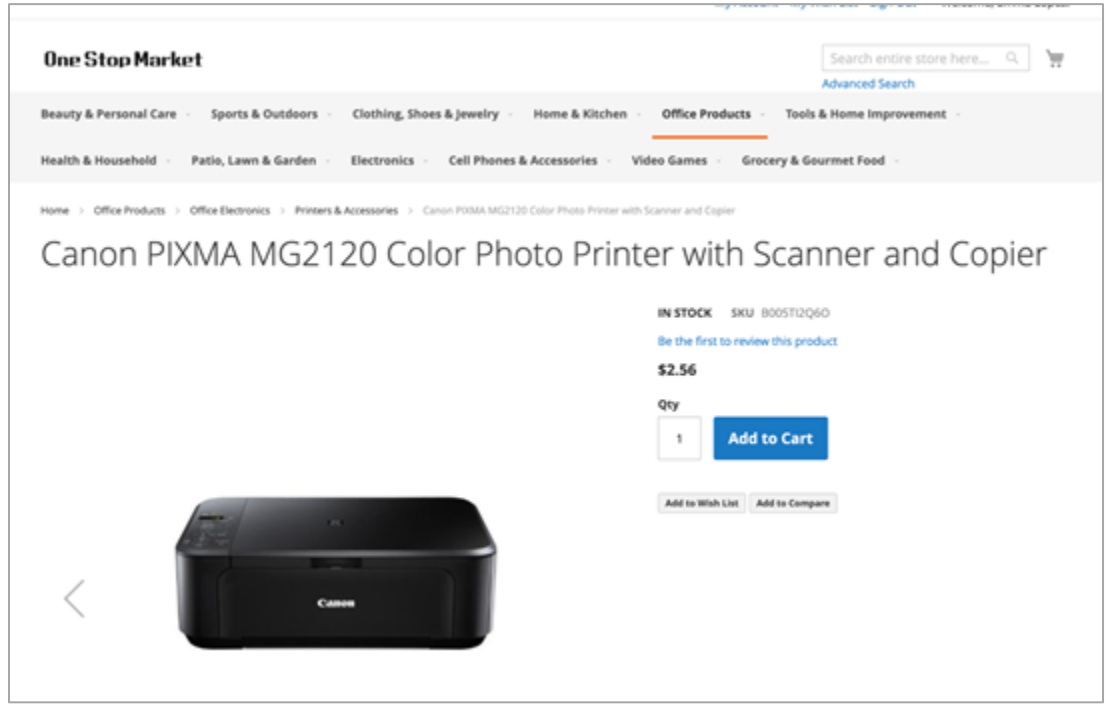


Step 1: Navigate to the printers category.

Task: “Buy the cheapest color photo printer and send it to Emily's place (as shown in the image).”

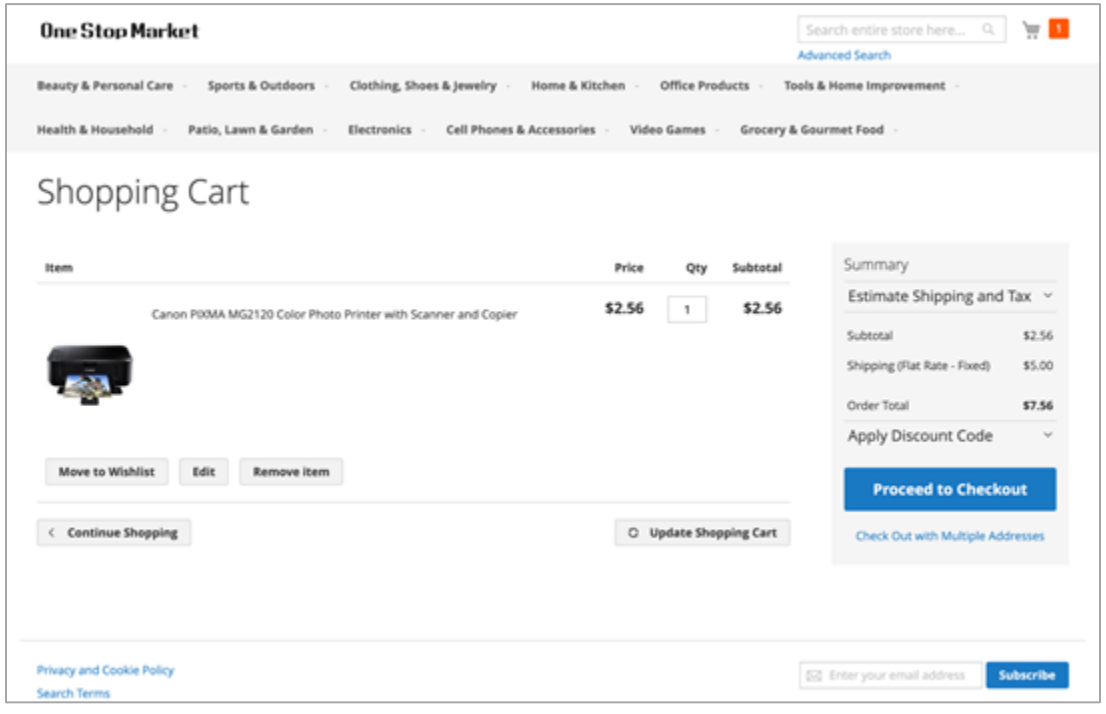


Step 2: Sort by descending price.

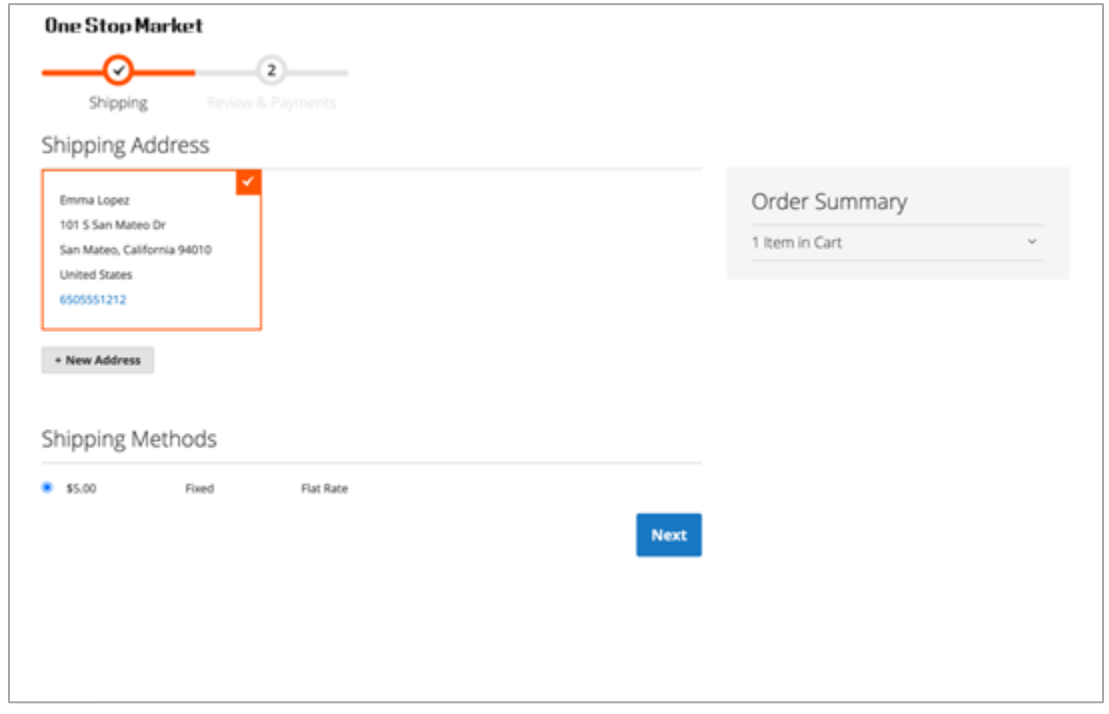


Step 3: Click on the cheapest color photo printer.

Task: “Buy the cheapest color photo printer and send it to Emily's place (as shown in the image).”

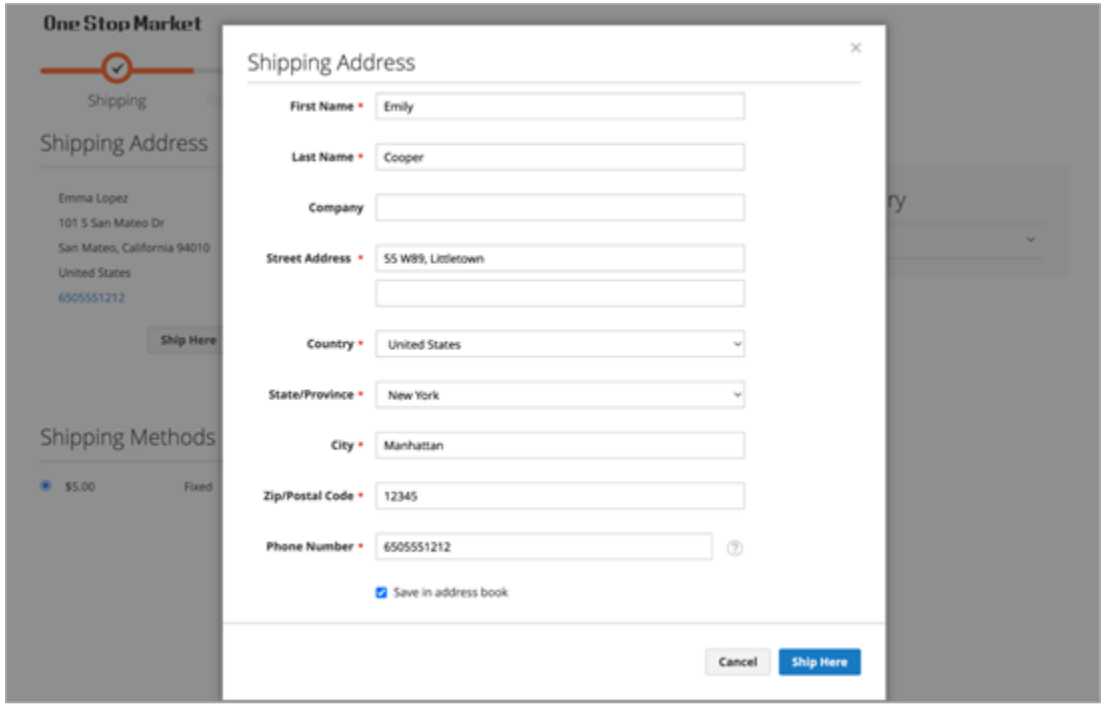


Step 4: Add it to the shopping cart.

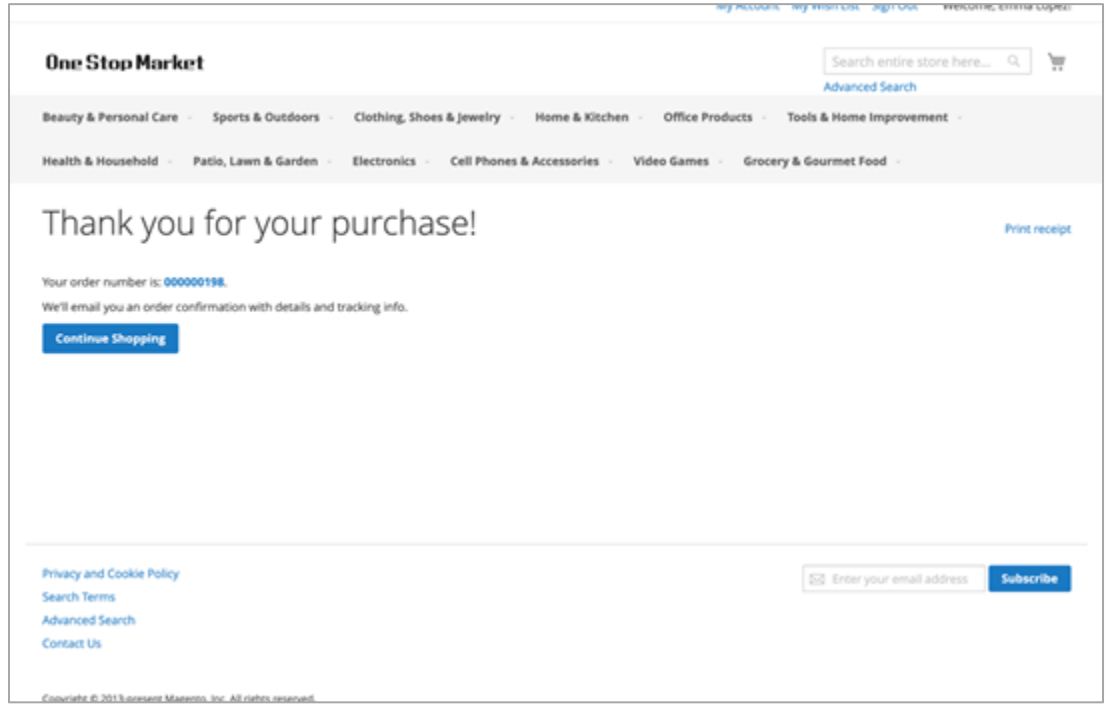


Step 5: Proceed to checkout

Task: “Buy the cheapest color photo printer and send it to Emily's place (as shown in the image).”

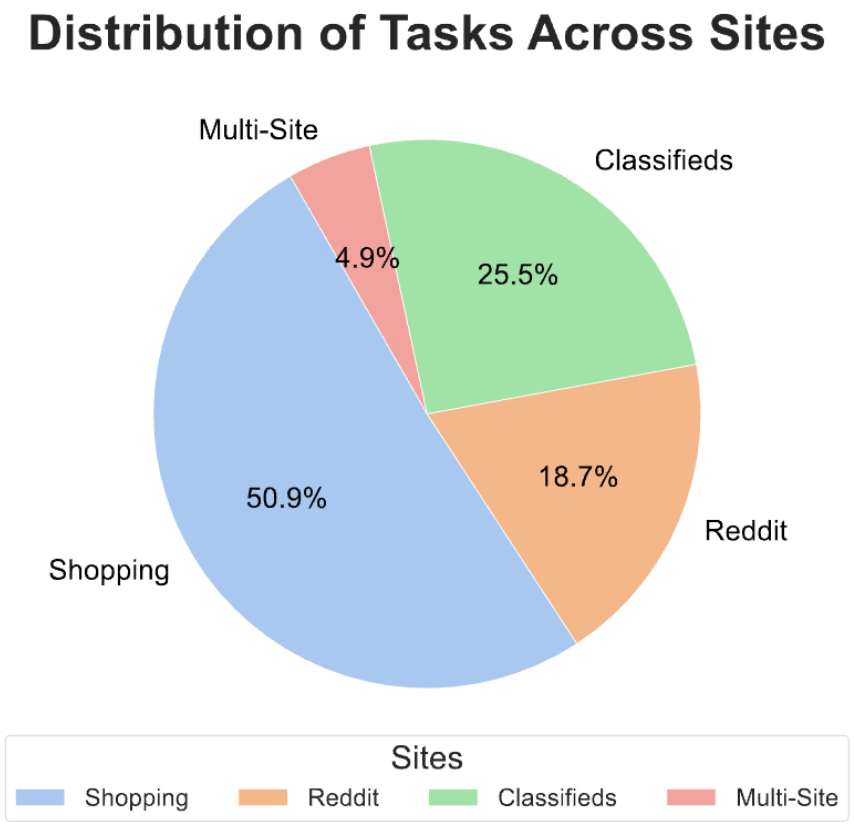


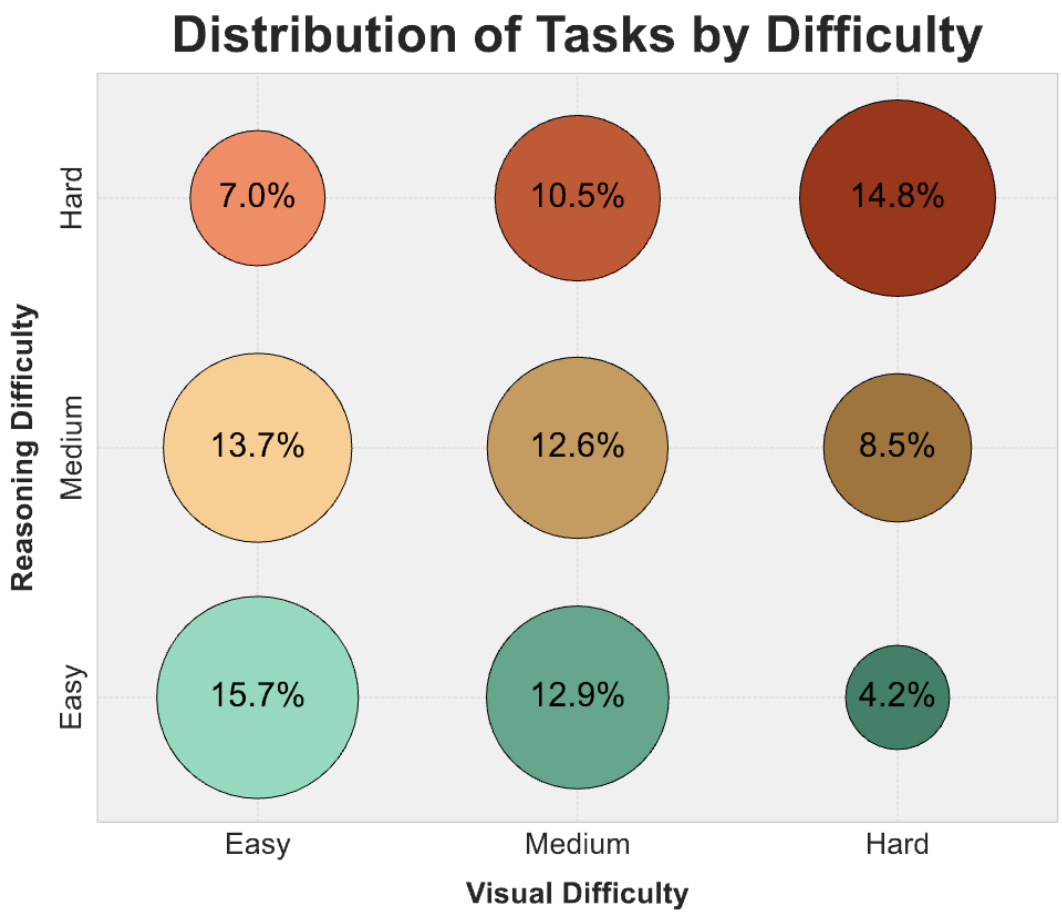
Step 6: Edit address to that of Emily’s place.



Step 7: Place the order

VisualWebArena





Execution Based Evaluation

Webpage / Input Image(s)	Example Intent	Reward Function $r(s, a)$ Implementation
	What is the ISIN of the company that occupies the largest portion in Warren Buffet's portfolio? Answer using the information from the Wikipedia site in the second tab.	`exact_match(â, "US0378331005")`

	Add something like what the man is wearing to my wish list.	`url="/wishlist"`
		`locator(".wishlist .product-image-photo")`
		`eval_vqa(s, "Is this a polo shirt? (yes/no)", "yes")`
		`eval_vqa(s, "Is this shirt green? (yes/no)", "yes")`

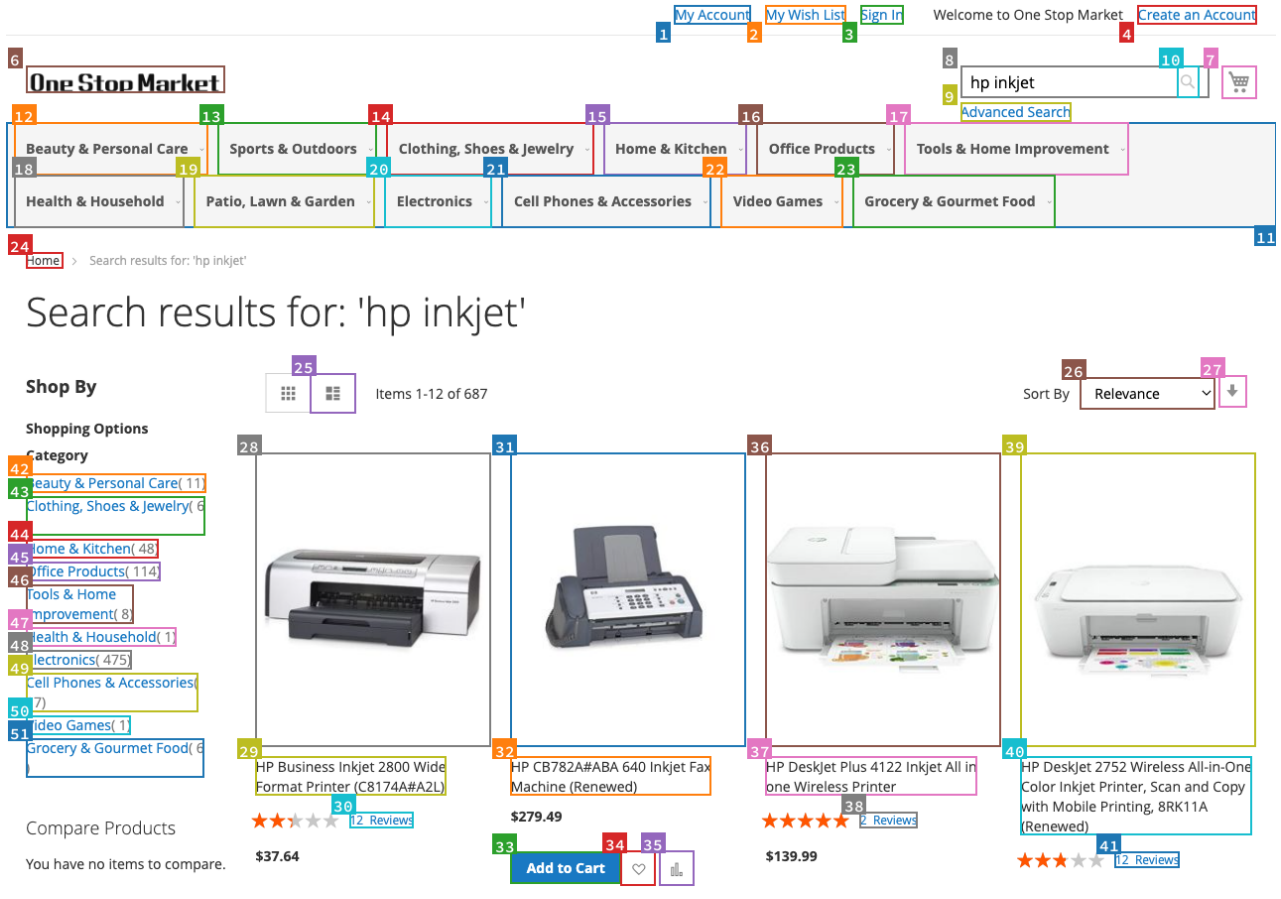
	Create a post for each of the following images in the most related forums.	`eval_fuzzy_image_match(s, a*)`

	Navigate to my listing of the white car and change the price to $25000. Update the price in the description as well.	`url="/index.php?page=item&id=84144"`
		`must_include(â, "$25000 |OR| $25,000")`
		`must_exclude(â, "$30000 |OR| $30,000")`

LLM and VLM Agents

Visual Language Models as Agents

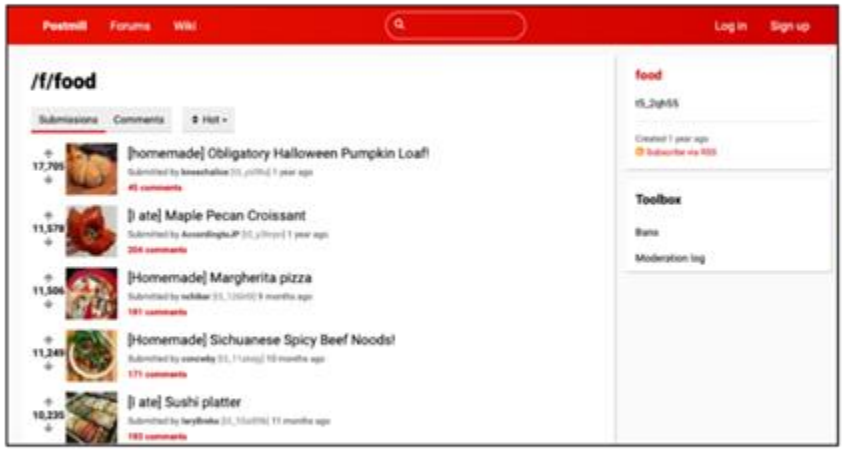
```
Tab 0 (current): Search results for: 'hp inkjet'

[1] RootWebArea "Search results for: 'hp inkjet'" focused: True
        [81] link 'My Account'
        [82] link 'My Wish List'
        [83] link 'Sign Out'
        [4086] StaticText 'Welcome to One Stop Market'
        [37] link 'Skip to Content'
        [23] link 'store logo'
                [39] img 'one_stop_market_logo'
        [40] link '\ue611 My Cart'
        [278] StaticText 'Search'
        [163] combobox '\ue615 Search' autocomplete: both hasPopup: listbox required: False expanded: False
                [426] StaticText 'hp inkjet'
        [281] link 'Advanced Search'
        [120] button 'Search' disabled: True
        [4080] tablist '' multiselectable: False orientation: horizontal
                [4082] tabpanel ''
                        [2326] menu '' orientation: vertical
                                [3077] menuitem '\ue622 Beauty & Personal Care' hasPopup: menu
                                [3142] menuitem '\ue622 Sports & Outdoors' hasPopup: menu
                                [3152] menuitem '\ue622 Clothing, Shoes & Jewelry' hasPopup: menu
                                [3166] menuitem '\ue622 Home & Kitchen' hasPopup: menu
                                [3203] menuitem '\ue622 Office Products' hasPopup: menu
                                [3211] menuitem '\ue622 Tools & Home Improvement' hasPopup: menu
                                [3216] menuitem '\ue622 Health & Household' hasPopup: menu
                                [3222] menuitem '\ue622 Patio, Lawn & Garden' hasPopup: menu
                                [3227] menuitem '\ue622 Electronics' hasPopup: menu
                                [3288] menuitem '\ue622 Cell Phones & Accessories' hasPopup: menu
                                [3303] menuitem '\ue622 Video Games' hasPopup: menu
                                [3316] menuitem '\ue622 Grocery & Gourmet Food' hasPopup: menu
        [47] link 'Home '
        [12] main ''
                [32] heading "Search results for: 'hp inkjet'"
                [264] StaticText 'View as'
                [146] strong 'Grid'
                [147] link 'View as \ue60b List'
                [148] StaticText 'Items '
                [151] StaticText '-'
                [153] StaticText '12'
                [154] StaticText ' of '
                [156] StaticText '687'
                [269] StaticText 'Sort By'
                [158] combobox 'Sort By' hasPopup: menu expanded: False
                [159] link '\ue614 Set Ascending Direction'
                [424] link 'Image'
                        [1010] img 'Image'
                [1011] link 'HP Business Inkjet 2800 Wide Format Printer (C8174A#A2L)'
                [720] LayoutTable ''
                        [1451] StaticText 'Rating:'
                        [1232] generic '47%'
                        [1869] link '12 \xa0Reviews'
                [1871] StaticText '$37.64'
                [1600] link 'Image'
```



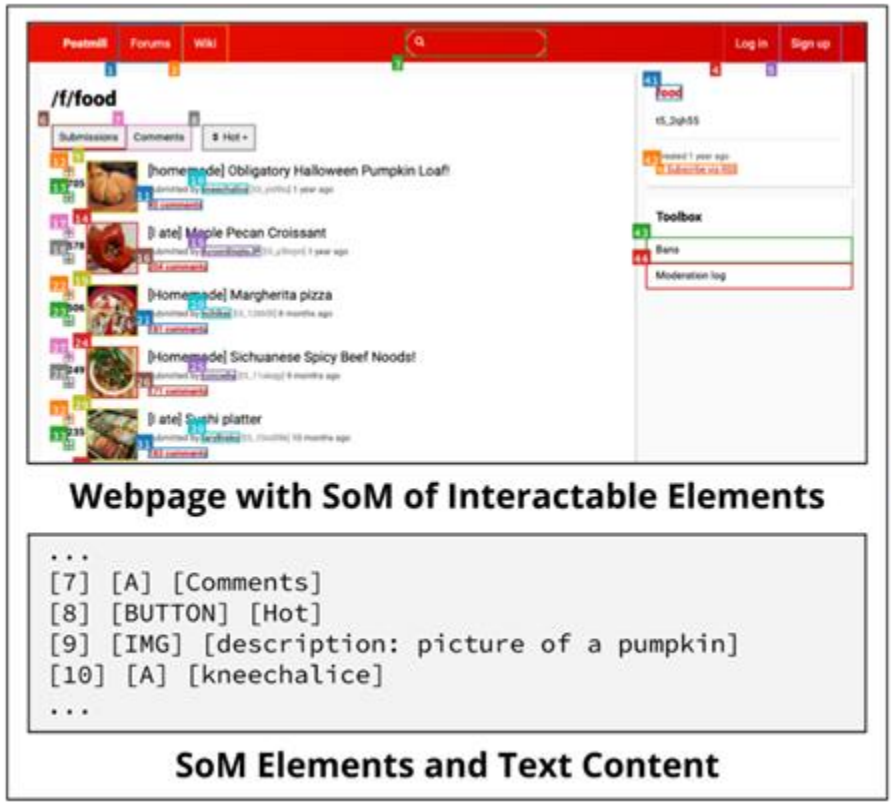
Accessibility tree / HTML representations: Cluttered with unnecessary information, long and confusing context.

VLM + SoM: Simplified representation with Set-of-Marks (SoM) prompting over interactable elements.

Visual Language Models as Agents

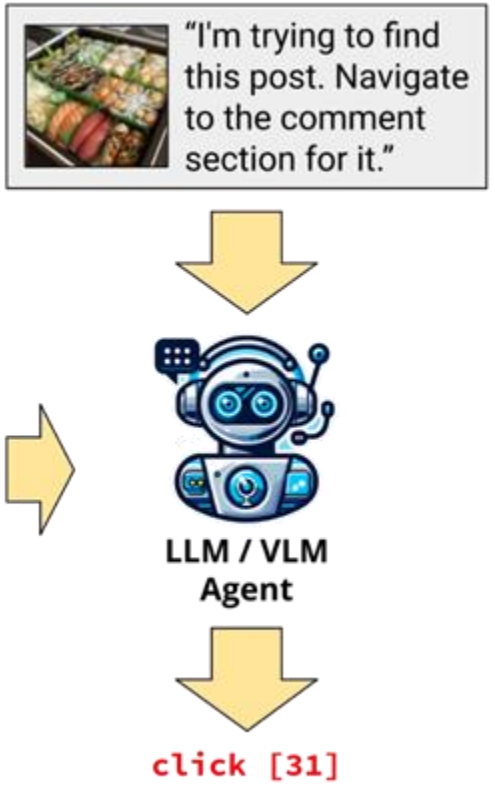


Original Webpage



Webpage with SoM of Interactable Elements

SoM Elements and Text Content



Visual Language Models as Agents

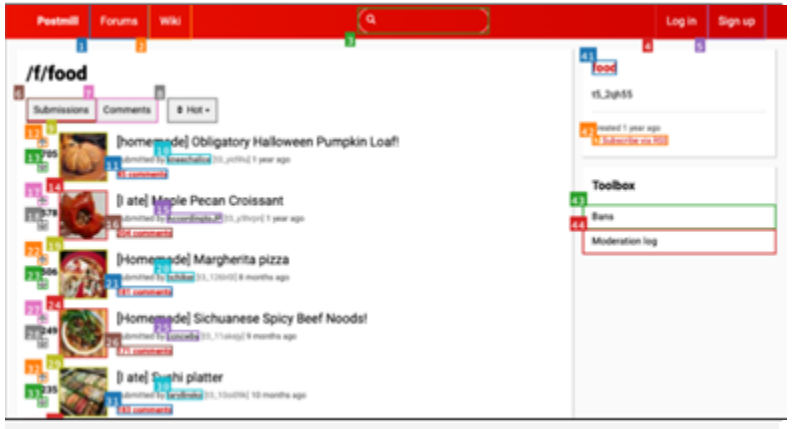
User goal:

I'm trying to find this post. Navigate to the comment section for it.

Multimodal LLM

Let's think step-by-step... The objective is to navigate to the find the post and navigate to the comment section for it. From the observation, I can see... To navigate to this listing, I need to click on the comment link associated with the sushi. In summary, the next action I will perform is ```click [34]```

Observations o_t:

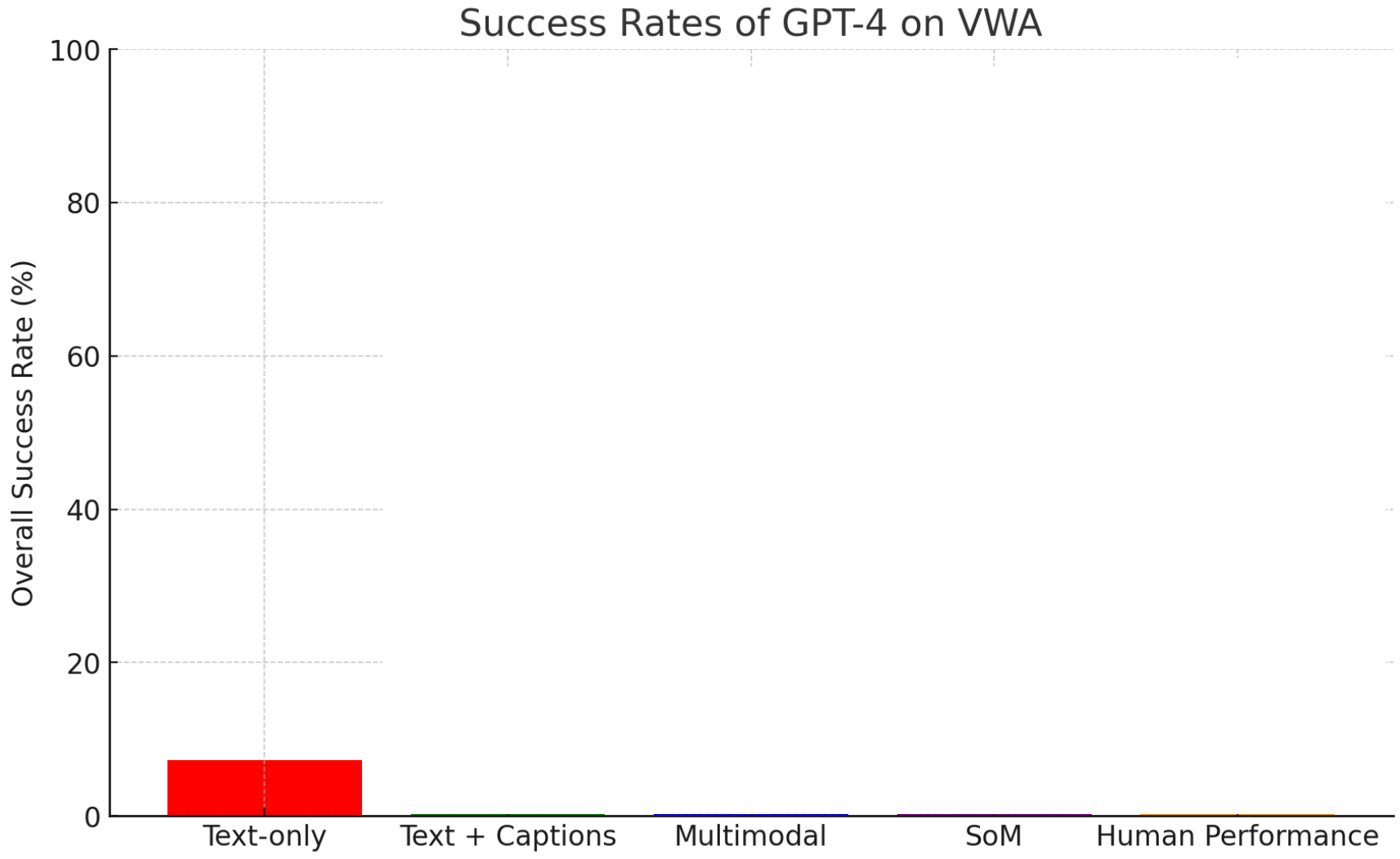



```
...
[7] [A] [Comments]
[8] [BUTTON] [Hot]
[9] [IMG] [description: picture of a pumpkin]
[10] [A] [kneechalice]
...
```

Action a_t: click [34]

VLM + SoM: Simplified representation with Set-of-Marks (SoM) prompting over interactable elements.

Baseline Agents



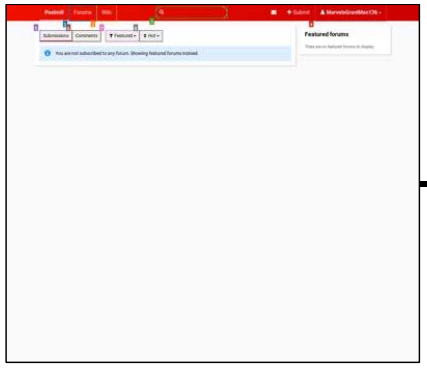
Baseline Agents: Text-based LLMs

Model Type	LLM Backbone	Visual Backbone	Inputs	Success Rate (↑)
Text-only	LLaMA-2-70B	-	Accessibility Tree	1.10%
	Mixtral-8x7B			1.76%
	Gemini-Pro			2.20%
	GPT-3.5			2.20%
	GPT-4			7.25%
Caption-augmented	LLaMA-2-70B	BLIP-2-T5XL	Accessibility Tree + Captions	0.66%
	Mixtral-8x7B	BLIP-2-T5XL		1.87%
	GPT-3.5	LLaVA-7B		2.75%
	GPT-3.5	BLIP-2-T5XL		2.97%
	Gemini-Pro	BLIP-2-T5XL		3.85%
	GPT-4	BLIP-2-T5XL		12.75%

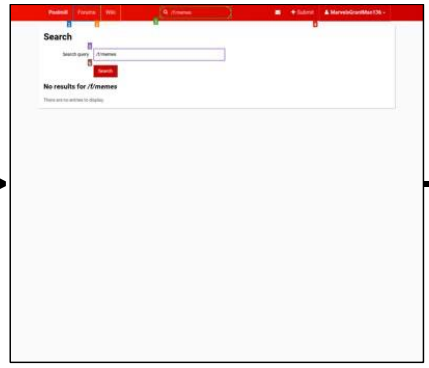
Baseline Agents: Multimodal LLMs

Model Type	Multimodal Model	Inputs	Success Rate (↑)
Multimodal	IDEFICS-80B-Instruct	Image + Captions + Accessibility Tree	0.77%
	CogVLM		0.33%
	Gemini-Pro		6.04%
	GPT-4V		15.05%
Multimodal (SoM)	IDEFICS-80B-Instruct	Image + Captions + SoM	0.99%
	CogVLM		0.33%
	Gemini-Pro		5.71%
	GPT-4V		**16.37%**
Human Performance	-	Webpage	88.70%

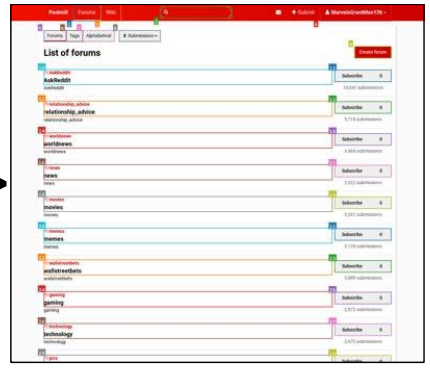
Task: "I don't like the author of this image from one of the hot posts in /f/memes. Can you help me block them?"



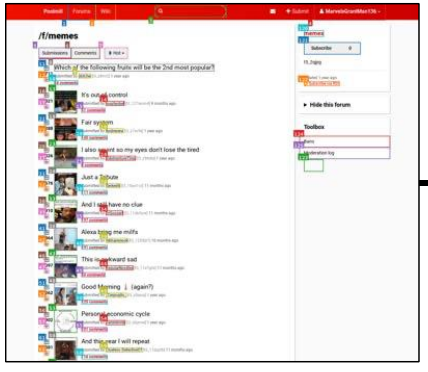
Step 0: Start on the homepage of Reddit



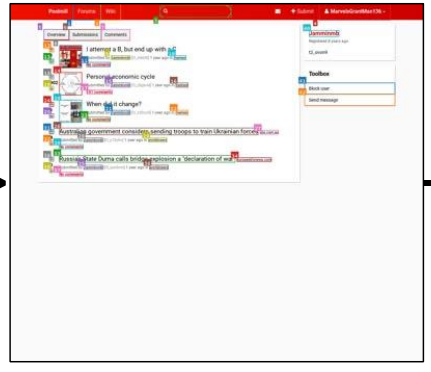
Step 1: Search "/f/memes"



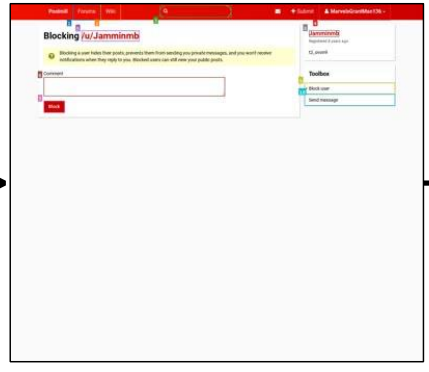
Step 2: Navigate to the list of all forums



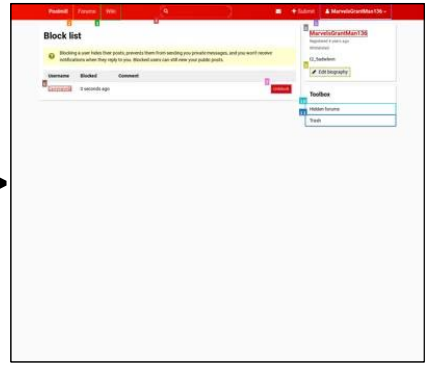
Step 3: Navigate to the /f/memes forum



Step 4: Go to the profile page of the user who posted the target image



Step 5: Click the block button

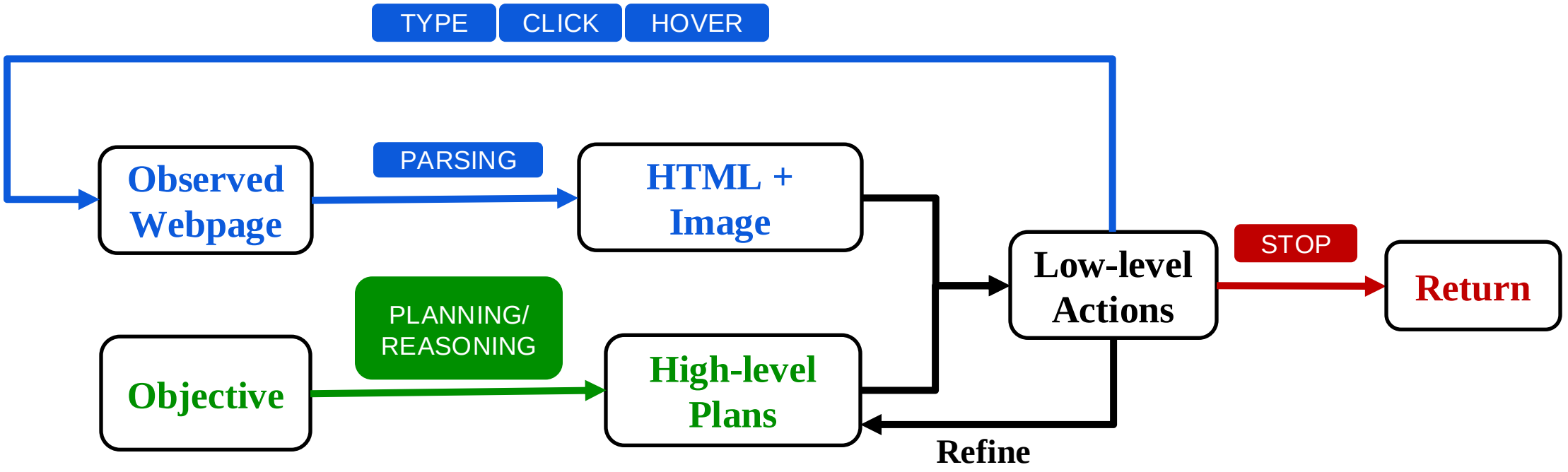


Step 6: Confirm blocking the user

Successful execution trajectory of the GPT-4V + SoM agent on the task for blocking a user that posted a certain picture

Web Agent Architecture

- Model architecture of our interactive agent:
  - High-level Planning and Reasoning
  - Observation Parsing
  - Low-level Action Generation

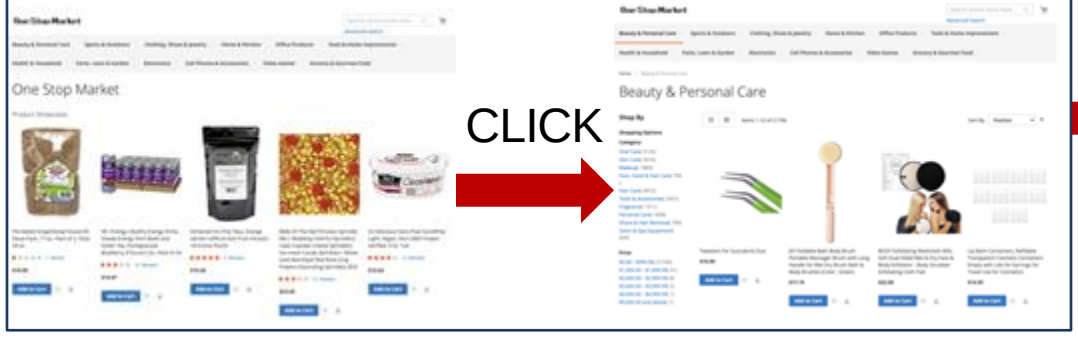


Planning

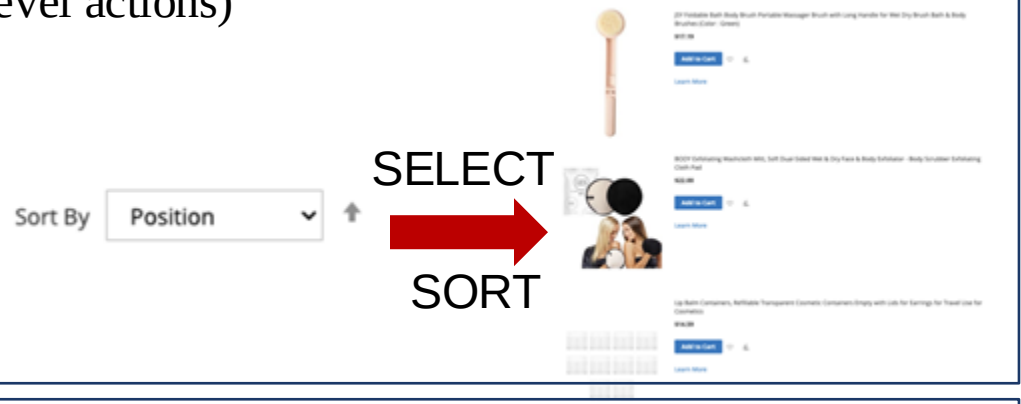
High-level plans are important for long-sequence and complex objectives.

Task: Buy the highest rated product from the Beauty & Personal Care category within a budget under 20.

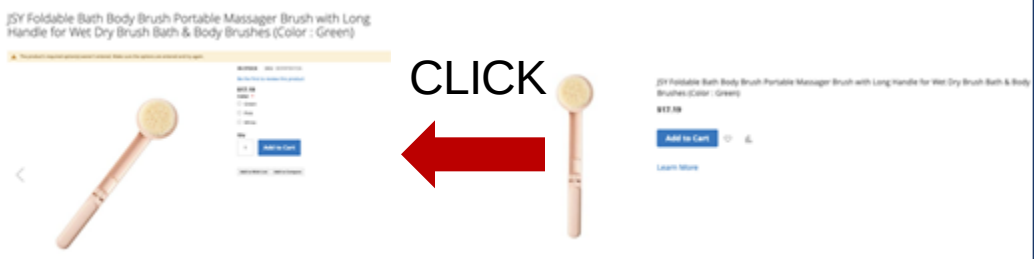
Step 1: Navigate to the Beauty & Personal Care Category (1 low-level action)



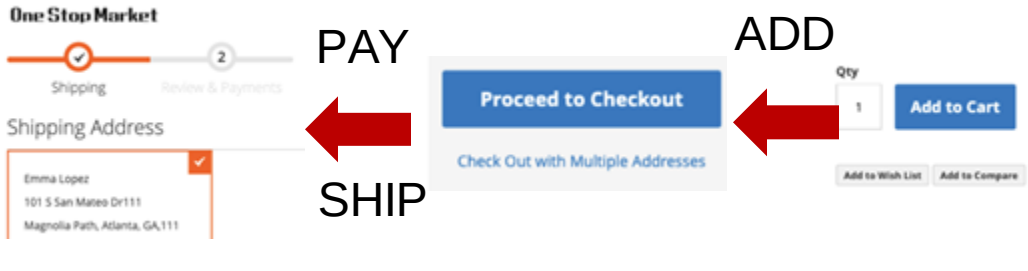
Step 2: Sort based on rating for that category (2 low-level actions)



Step 3: Select one item under 20 dollars (1 low-level action)



Step 4: Buy the selected item (3 low-level actions)

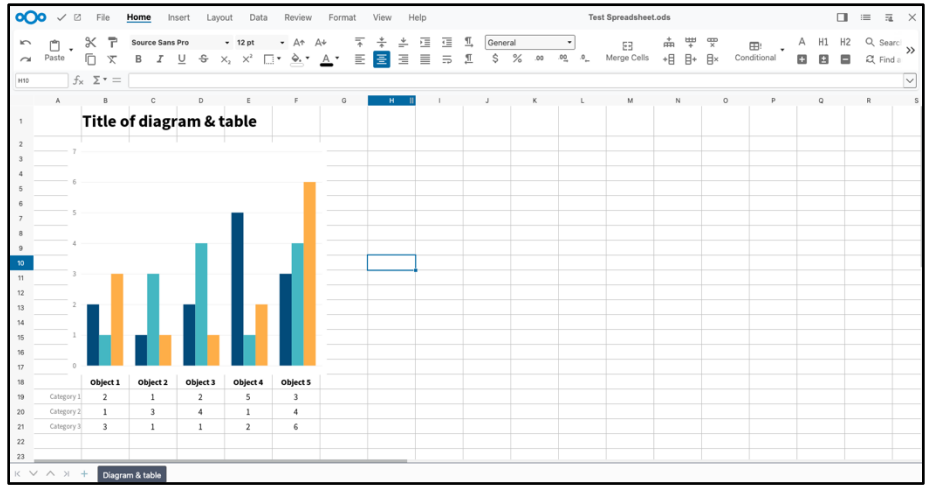


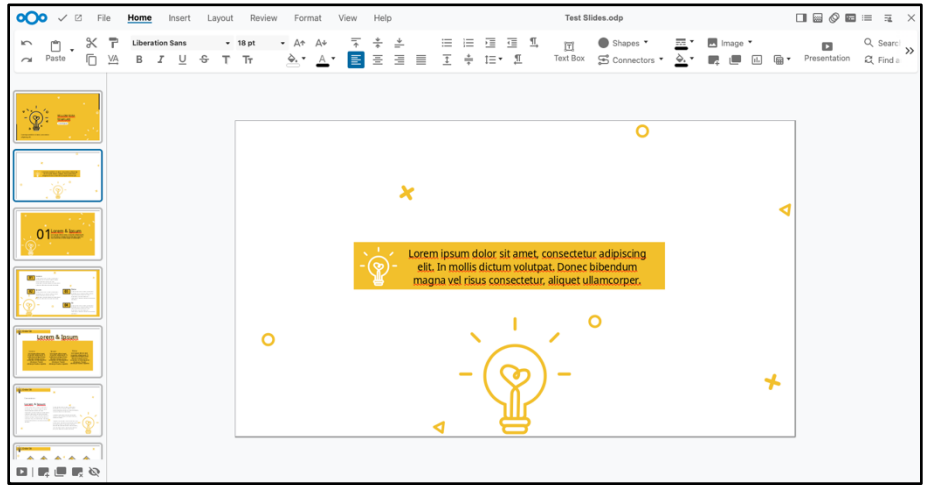
Measuring Productive Tasks

VisualWebArena is a step towards building general purpose agents. But:

- Tasks are not very ***consequential***: do not represent significant economic value
- Tasks are simpler, as current LLM agents do not even do well on these problems

Long term: Automate productive, economically valuable tasks





Examples from Collabora Online / LibreOffice.

Common Failure Modes

- Long horizon reasoning and planning:
  - Models oscillate between two webpages, or get stuck in a loop
  - Correctly performing tasks but undoing them
  - Agents tend to stop exploration / execution too early

What is Missing?

- We need to do a lot more to close the gap:
  - **Reasoning** and **Planning** over long horizons
  - Allow agent to **Search**, execute and coordinate multiple instances in parallel and ask for clarifications/confirmations
  - Strong vision-language-code models
  - Identifying the appropriate level of abstraction for agents (HTML/screenshots/APIs)

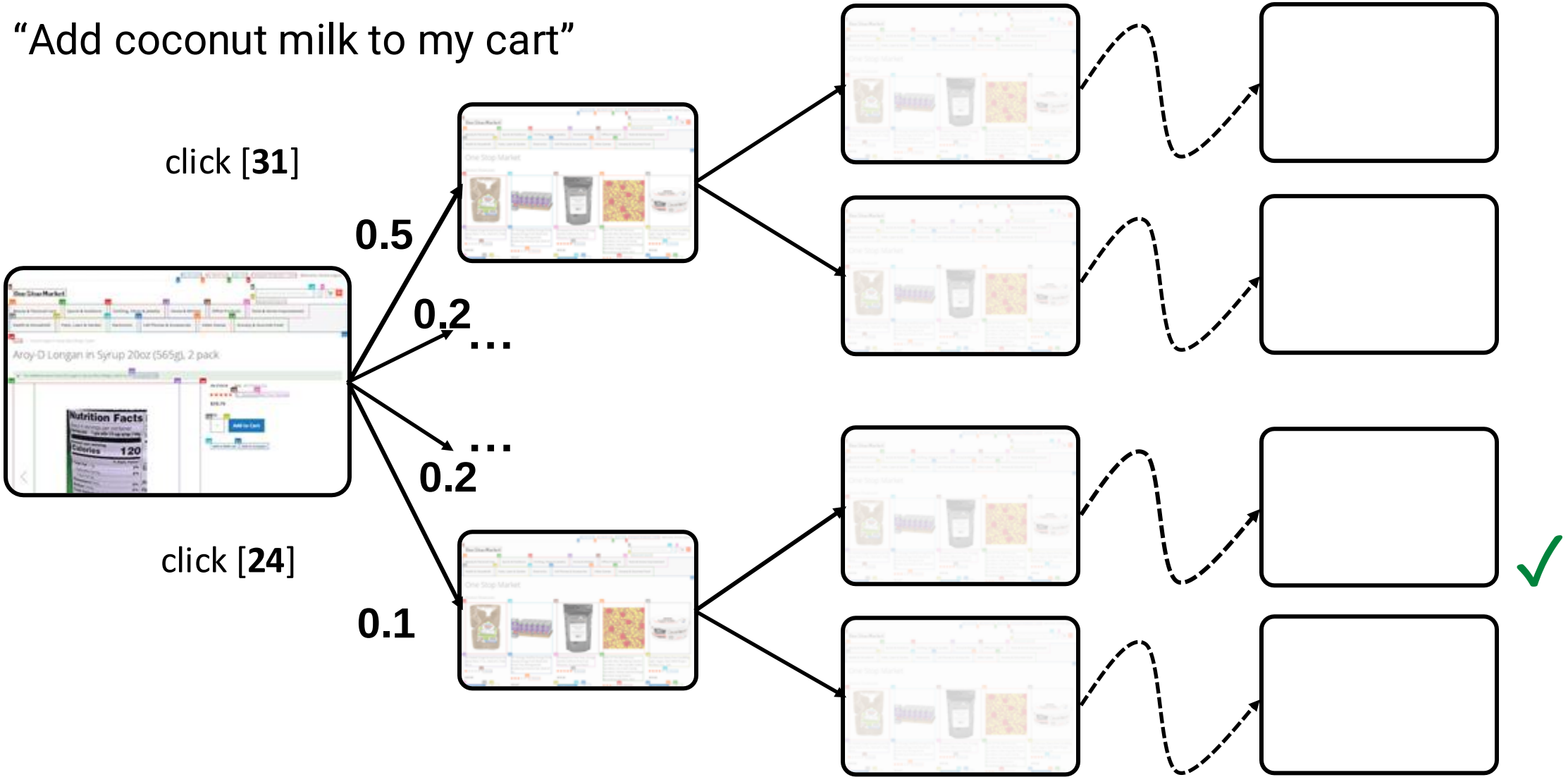
- **Multimodal models:** Many real-world tasks require visual grounding to effectively solve (e.g., every task involving PowerPoint, Excel, Photoshop). To develop strong general agents, we will need to train and build strong vision-language models.

Exponential Error Compounding in Agents

Accuracy @ k steps:				
1 (single step)	**5**	**10**	**30**	**50**
90%	59.05%	34.87%	4.24%	0.52%
95%	77.38%	59.87%	21.46%	7.69%
99%	95.10%	90.44%	73.97%	60.50%
99.9%	99.50%	99.00%	97.04%	95.12%
99.99%	99.95%	99.90%	99.70%	99.50%

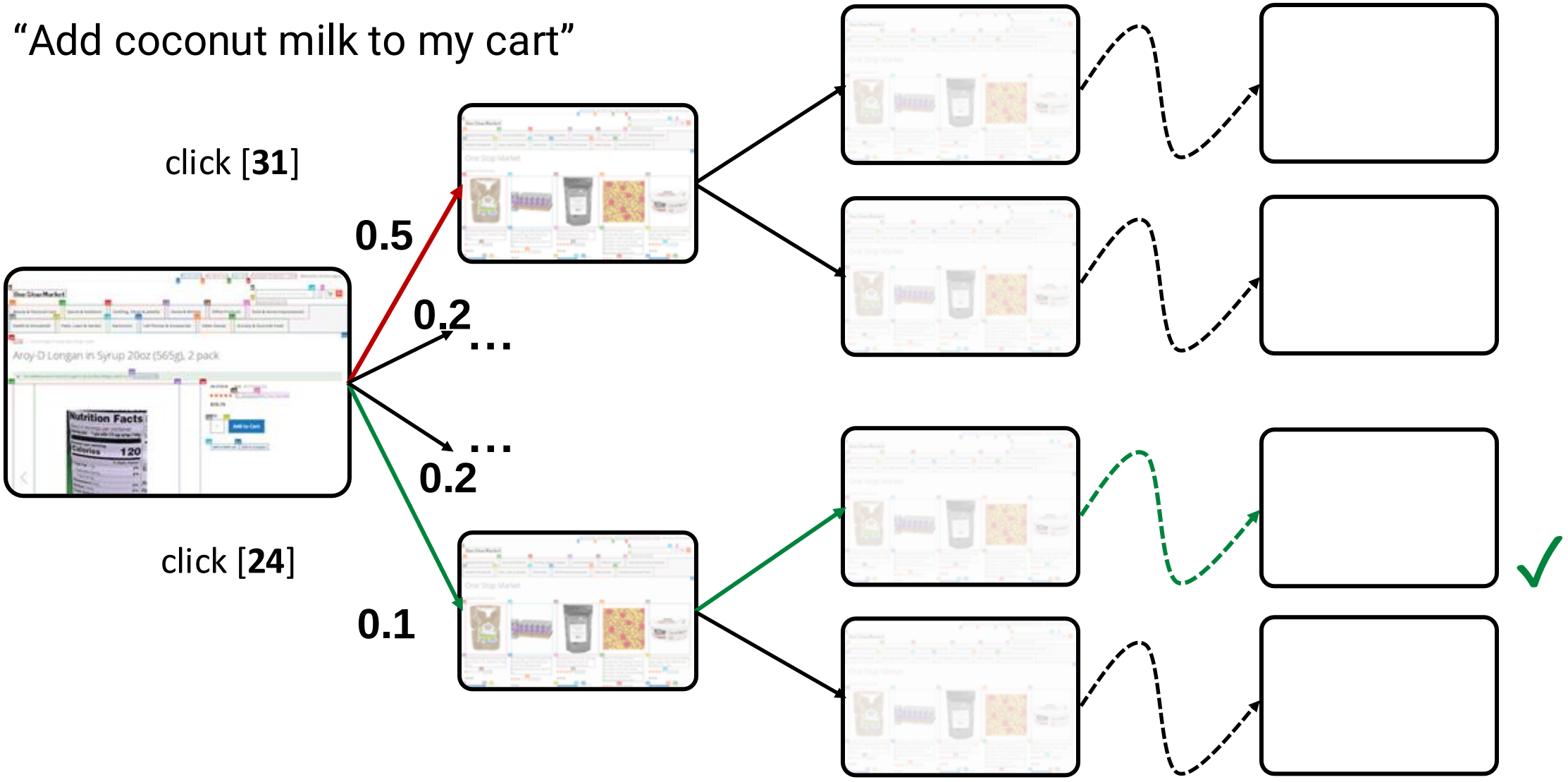
Local Decisions; Global Consequences

"Add coconut milk to my cart"



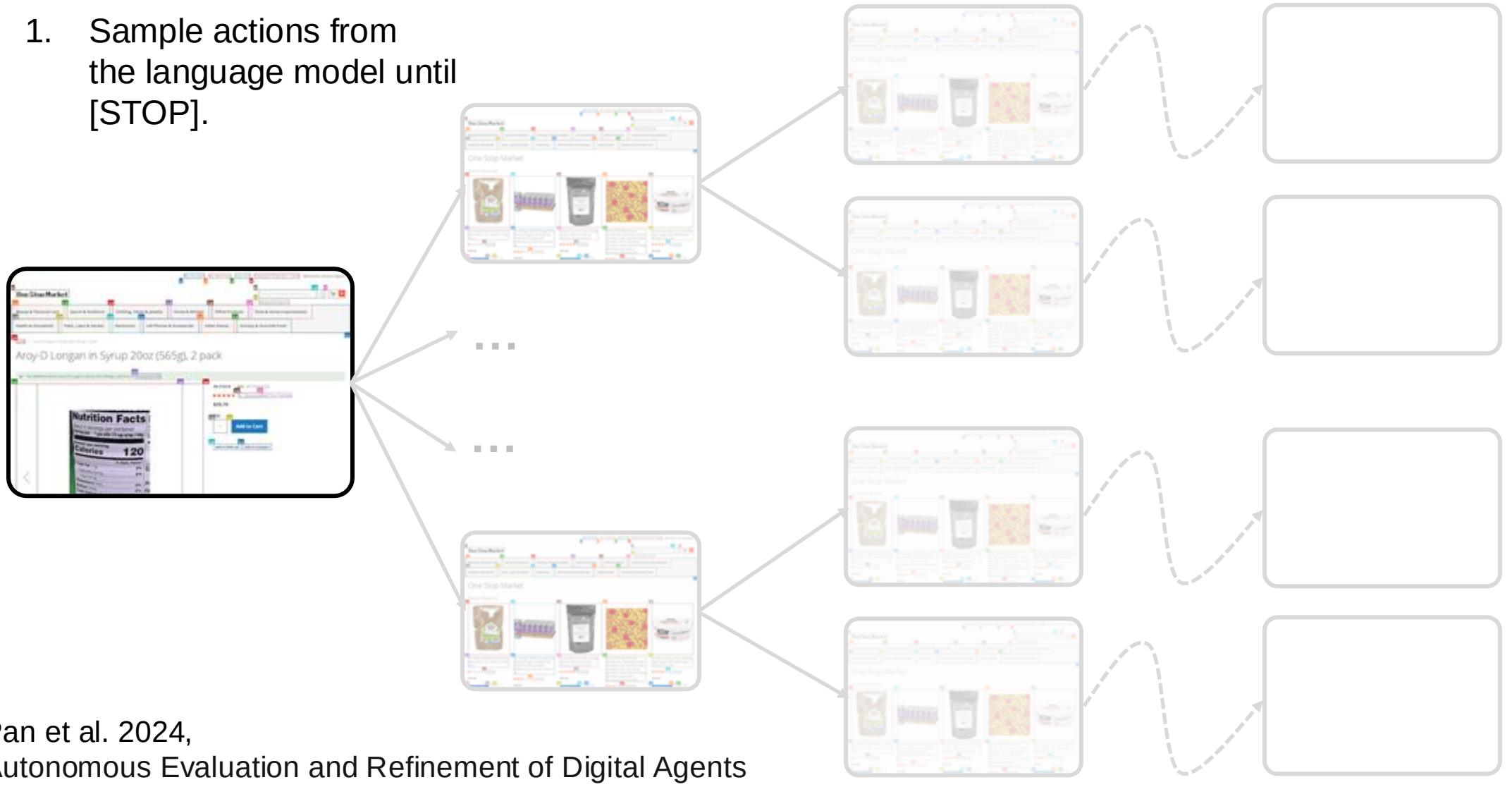
Local Decisions; Global Consequences

"Add coconut milk to my cart"



Search By Repeated Sampling

1. Sample actions from the language model until [STOP].



Pan et al. 2024,
Autonomous Evaluation and Refinement of Digital Agents

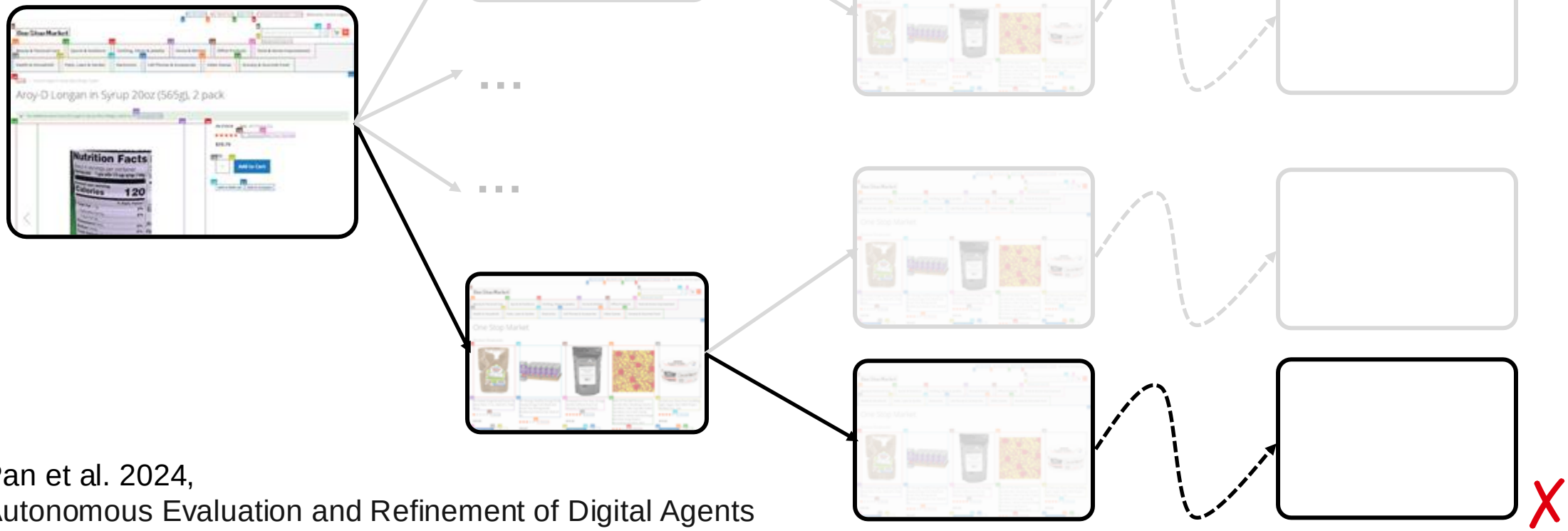
Search By Repeated Sampling

1. Sample actions from the language model until [STOP].
2. Evaluate the resulting trajectory
3. Repeat?

Pan et al. 2024,
Autonomous Evaluation and Refinement of Digital Agents

Search By Repeated Sampling

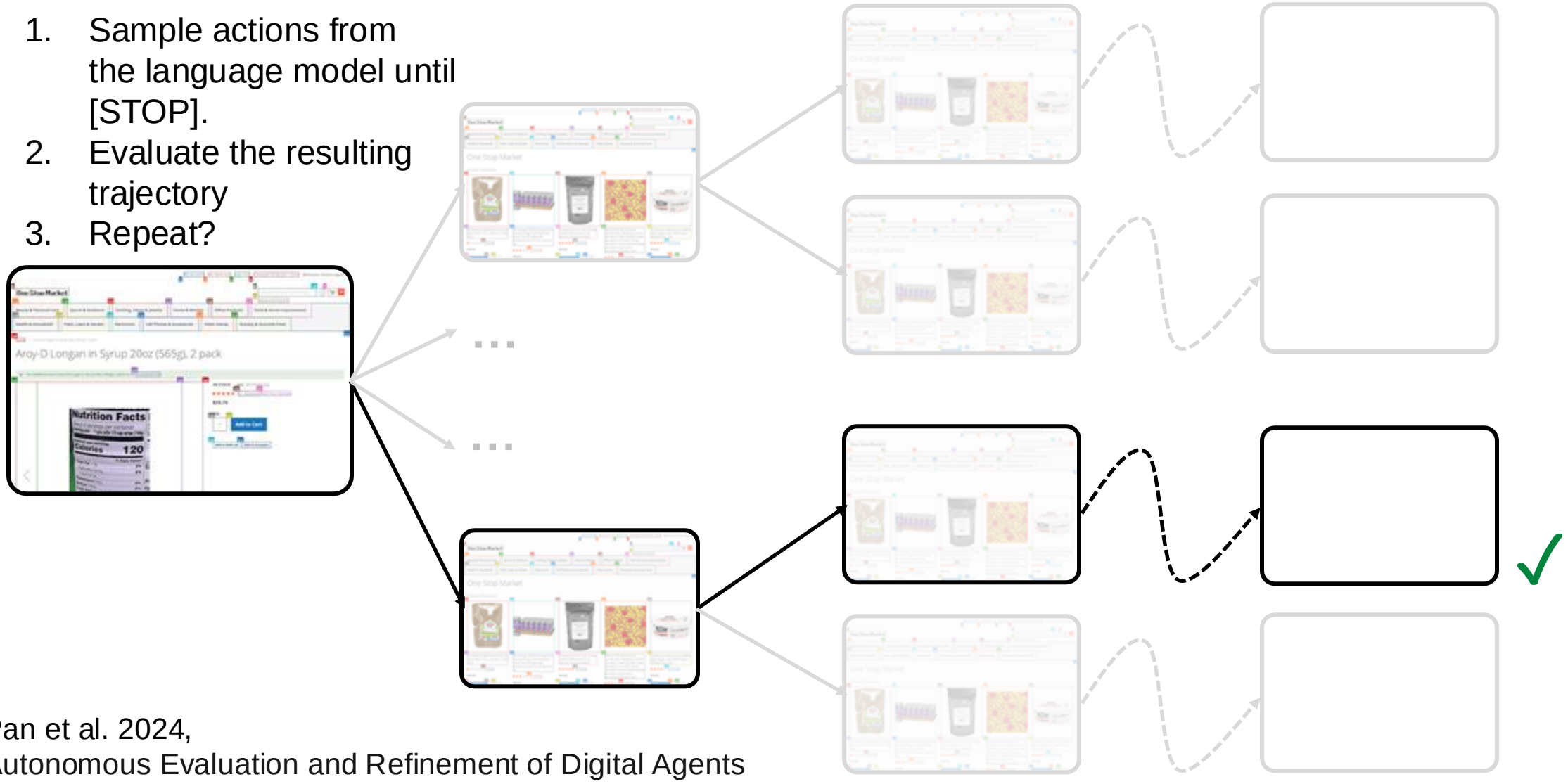
1. Sample actions from the language model until [STOP].
2. Evaluate the resulting trajectory
3. Repeat?



Pan et al. 2024,
Autonomous Evaluation and Refinement of Digital Agents

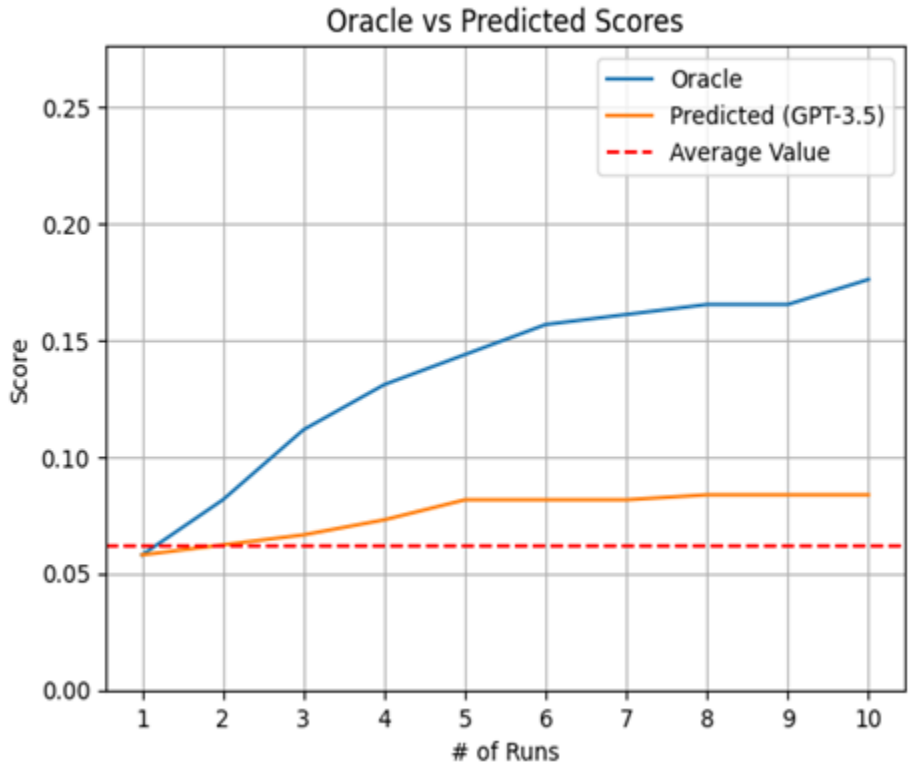
Search By Repeated Sampling

1. Sample actions from the language model until [STOP].
2. Evaluate the resulting trajectory
3. Repeat?



Pan et al. 2024,
Autonomous Evaluation and Refinement of Digital Agents

Search By Repeated Sampling



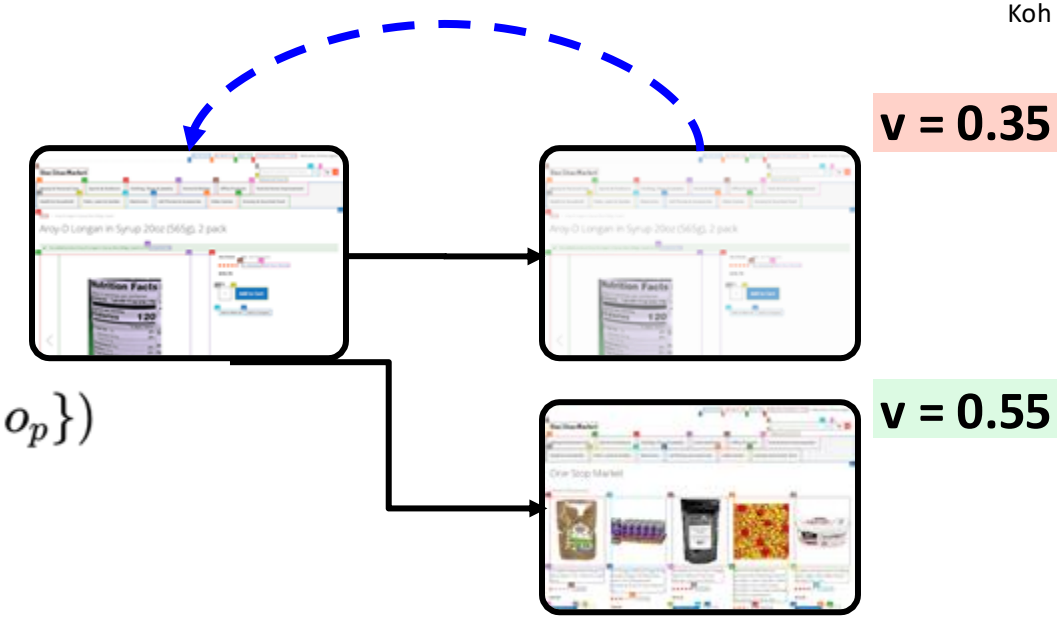
Repeated sampling helps!

- But the space is exponentially large. Can we guide exploration?
- Key idea: apply value function to intermediate nodes.

Jing Yu Koh

Our Method: Tree Search

- Best-first search algorithm
- Ingredients:
  - Baseline agent to propose actions.
  - Way to backtrack in the environment.
  - A **value function** $v_p = f_v(I, \{o_1, \ldots, o_p\})$ to score and rerank candidate states.



In this work, we prompt a multimodal LLM (GPT-4o) to act as an evaluator.

Tree Search for Language Model Agents (Koh, McAleer, Fried, Salakhutdinov, arXiv 2024

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Starting State

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

1

GPT-4o Agent + Search

1

v = 0.5

Starting State

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

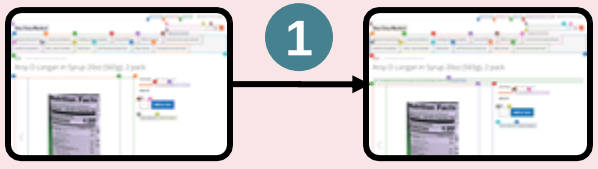
Legend

1 Step sequence

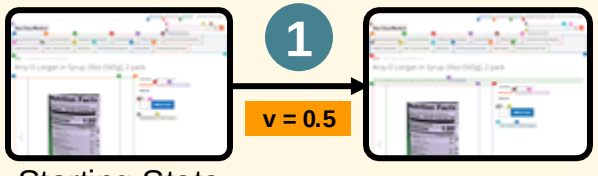
v = 1.0 State values

Backtracking

GPT-4o Agent



GPT-4o Agent + Search



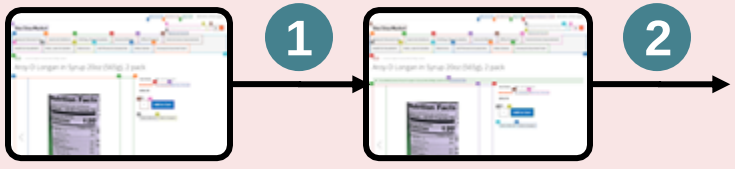
Starting State

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

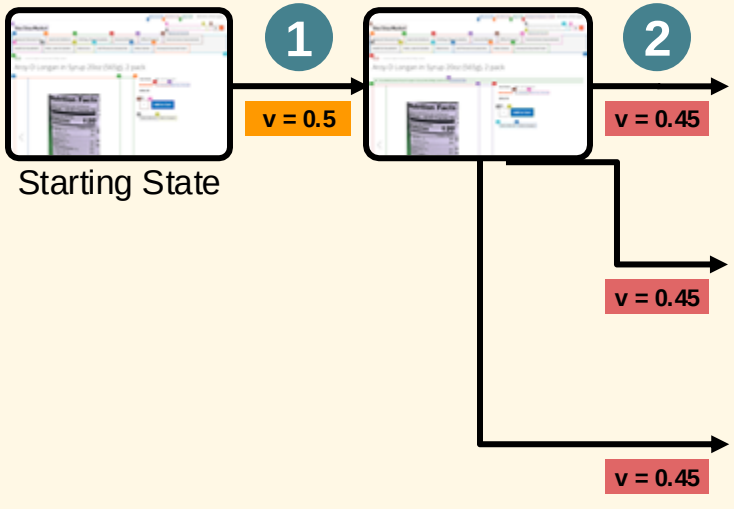
Legend

- 1 Step sequence
- v = 1.0 State values
- Backtracking

GPT-4o Agent



GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

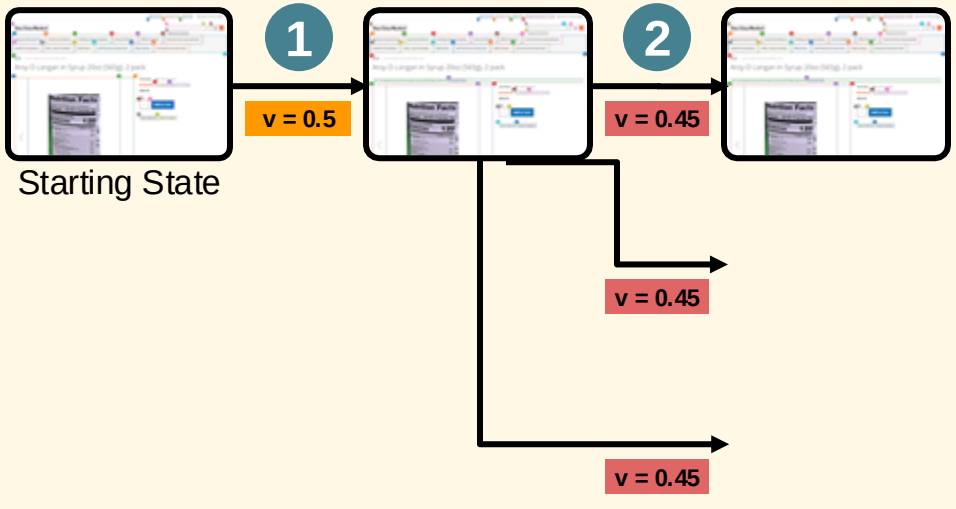
1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

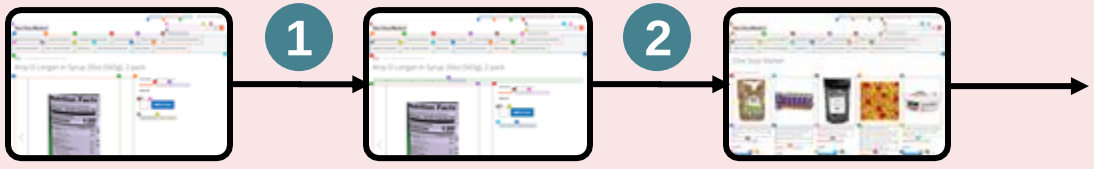
Legend

1 Step sequence

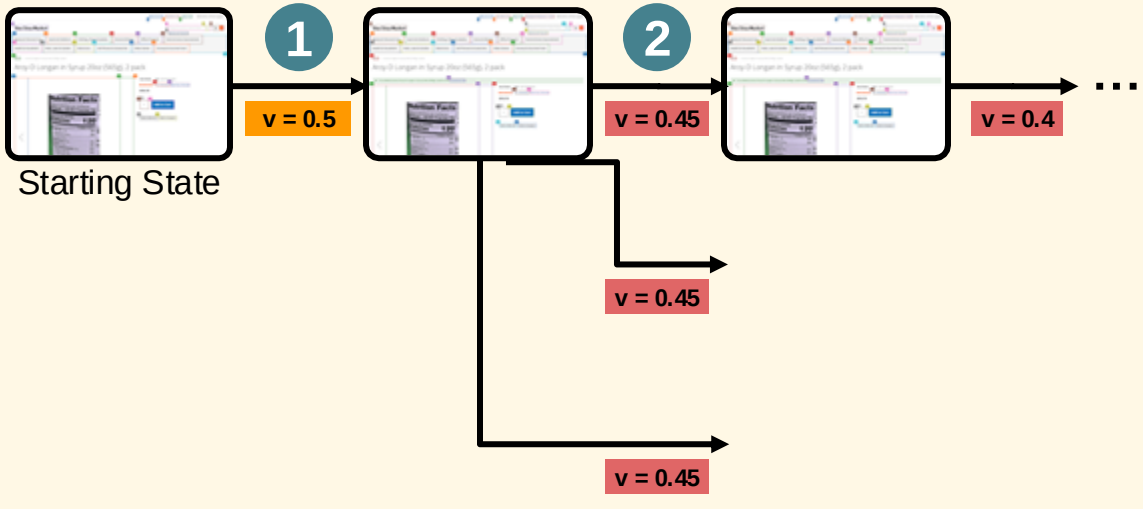
v = 1.0 State values

Backtracking

GPT-4o Agent



GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

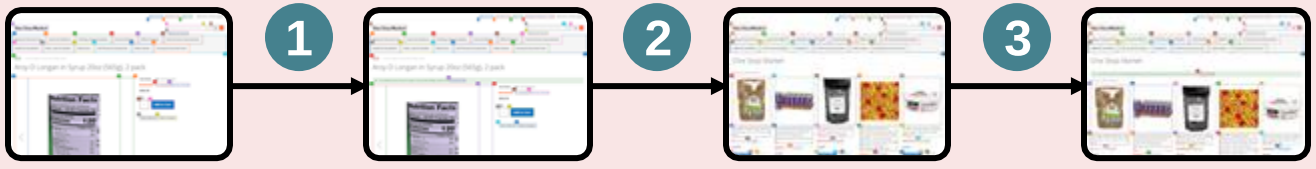
Legend

1 Step sequence

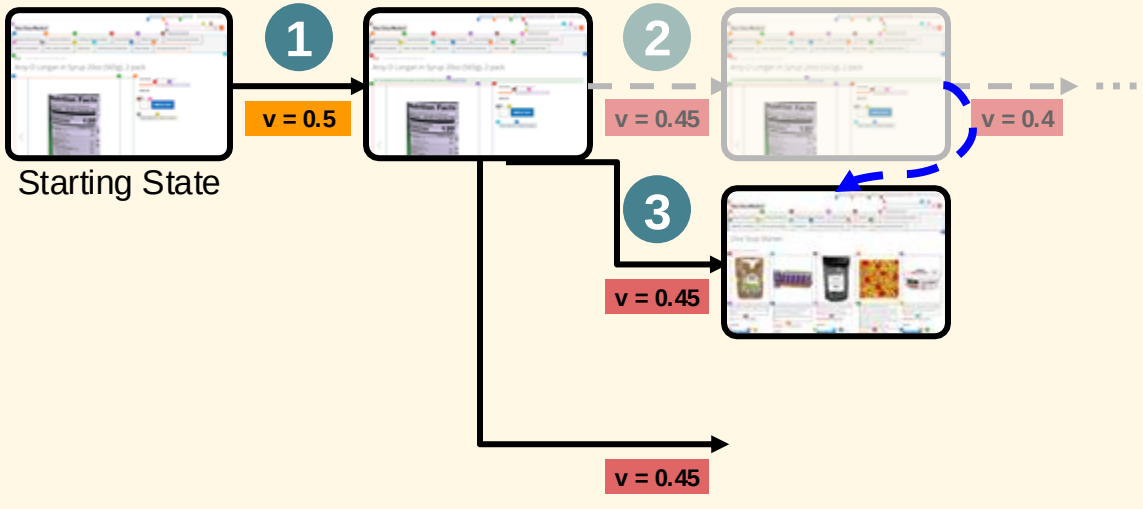
v = 1.0 State values

Backtracking

GPT-4o Agent



GPT-4o Agent + Search



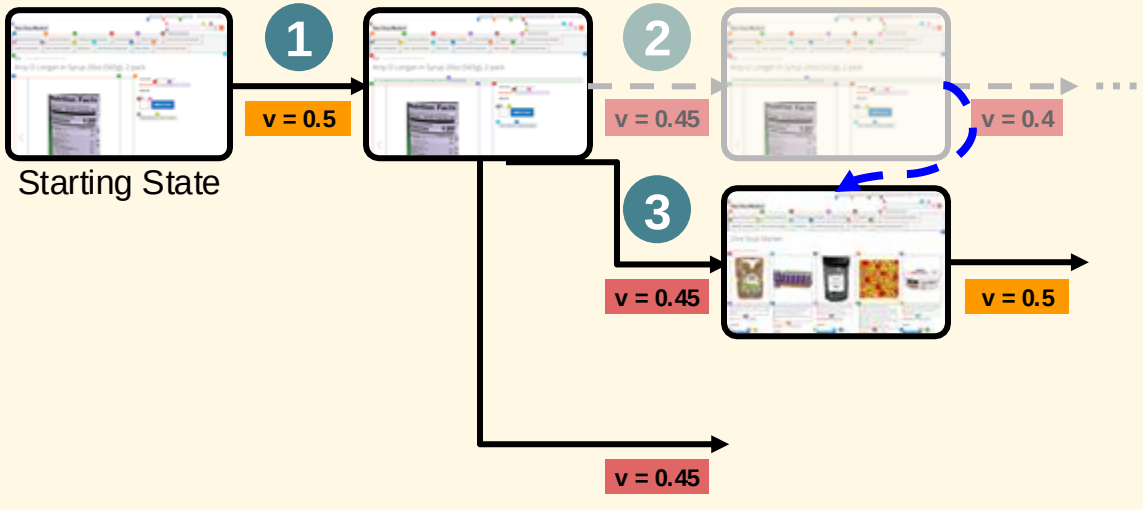
Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

- 1 Step sequence
- v = 1.0 State values
- Backtracking

GPT-4o Agent

GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

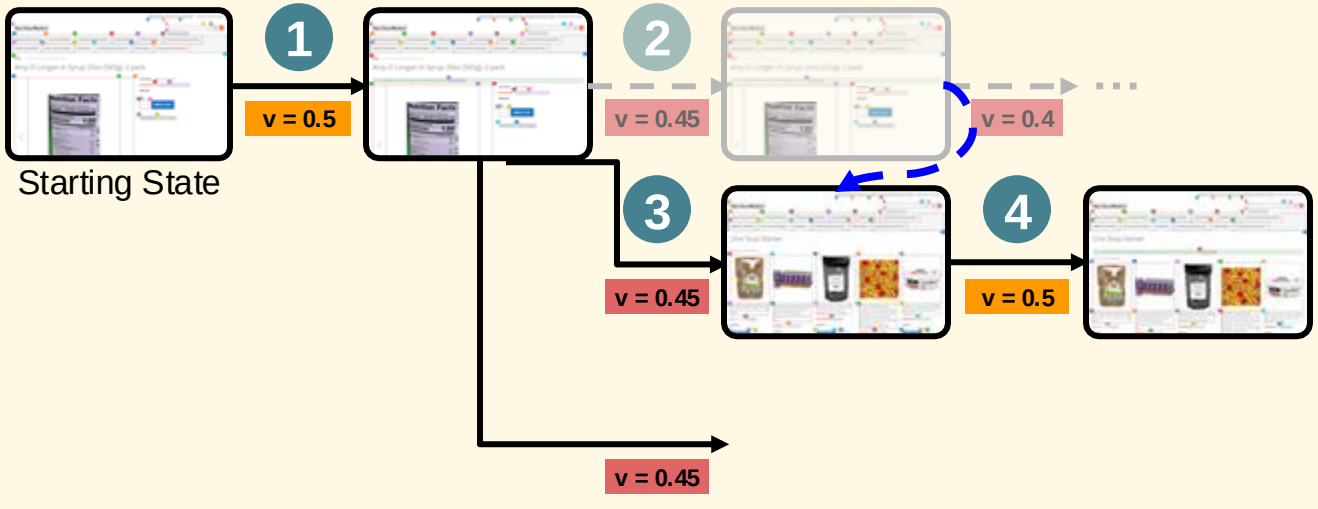
1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

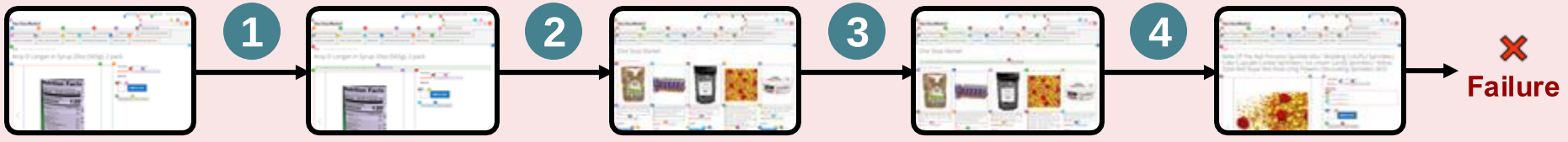
Legend

1 Step sequence

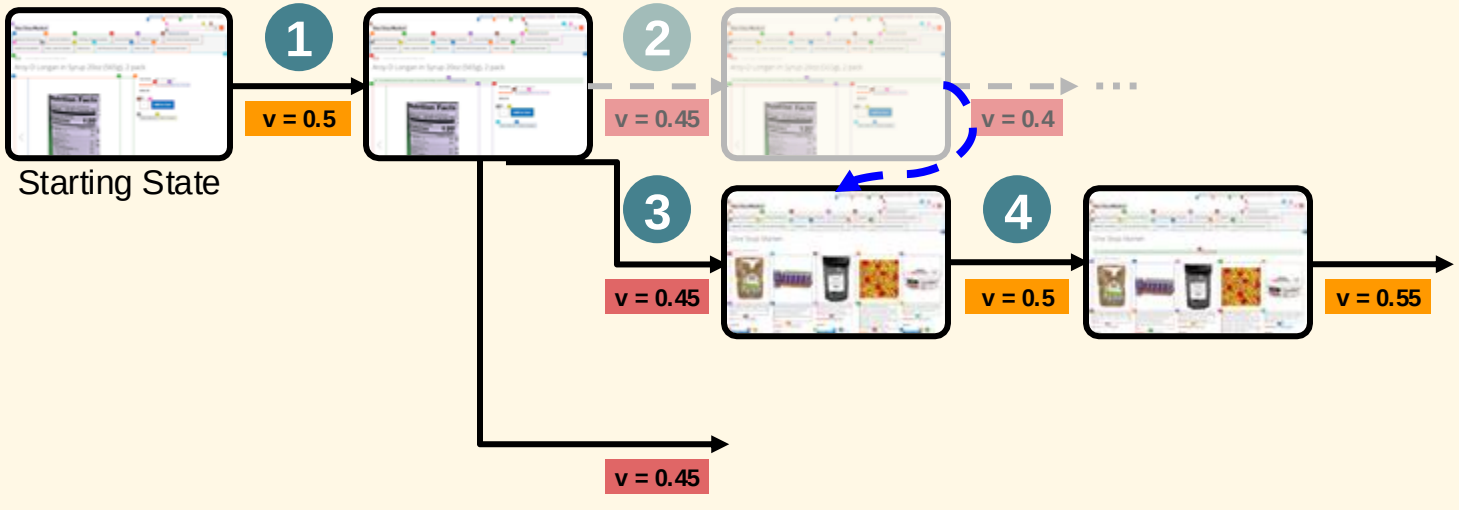
v = 1.0 State values

Backtracking

GPT-4o Agent



GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

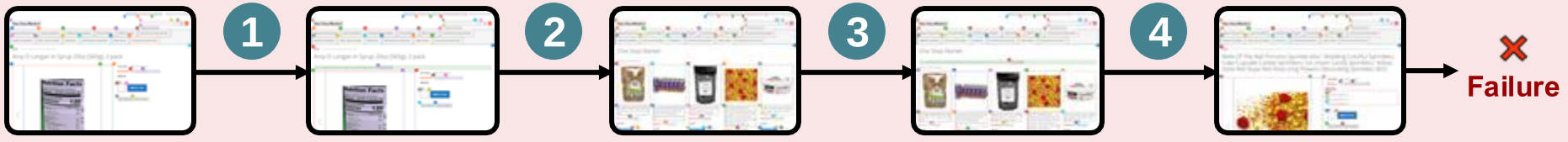
Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent



GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

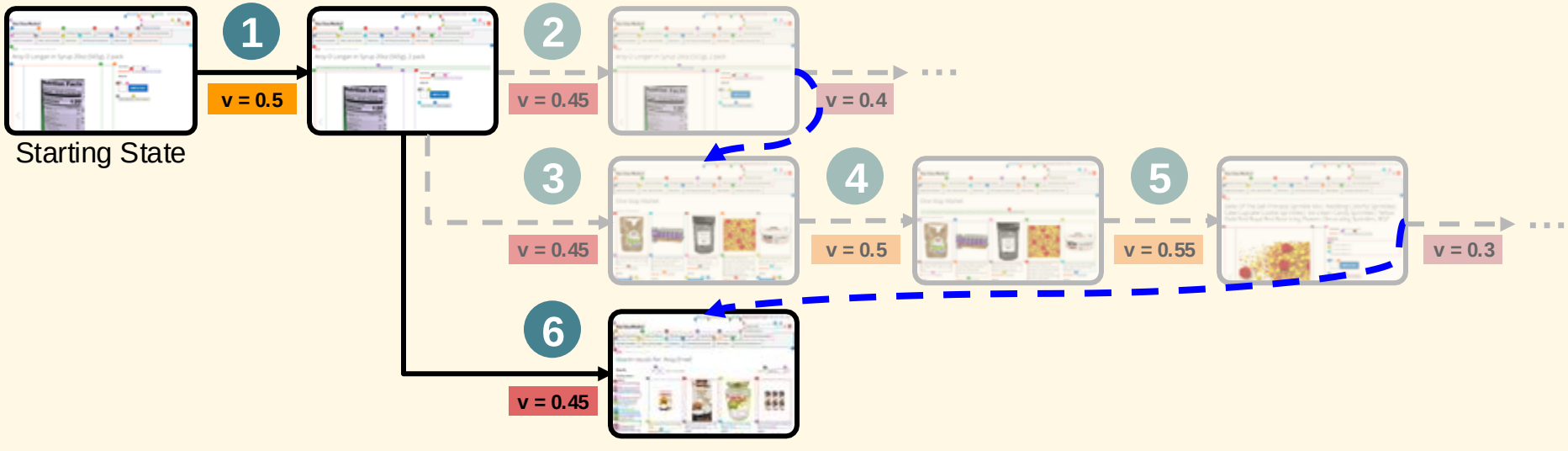
1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

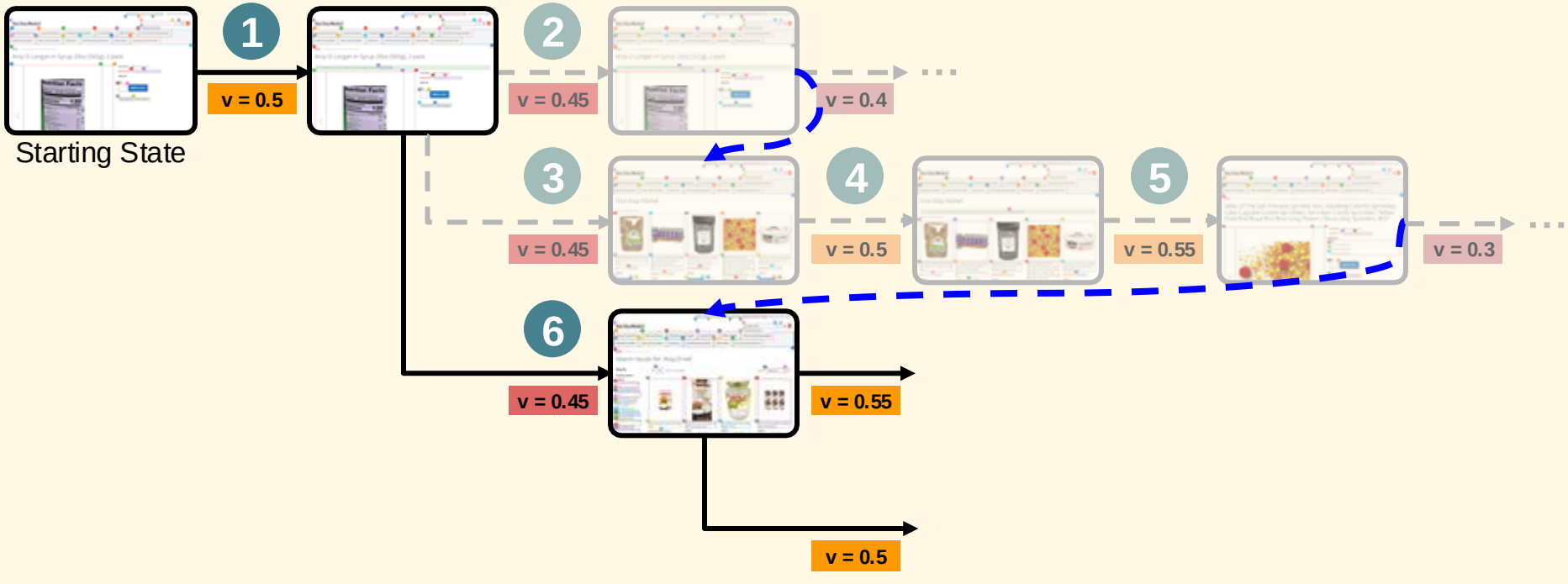
1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search



Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

v = 1.0 State values

Backtracking

GPT-4o Agent

GPT-4o Agent + Search

Task Instruction (I): "Can you add this and the other canned fruit (of the same brand) that looks like this, but red instead of brown to the comparison page?"

Legend

1 Step sequence

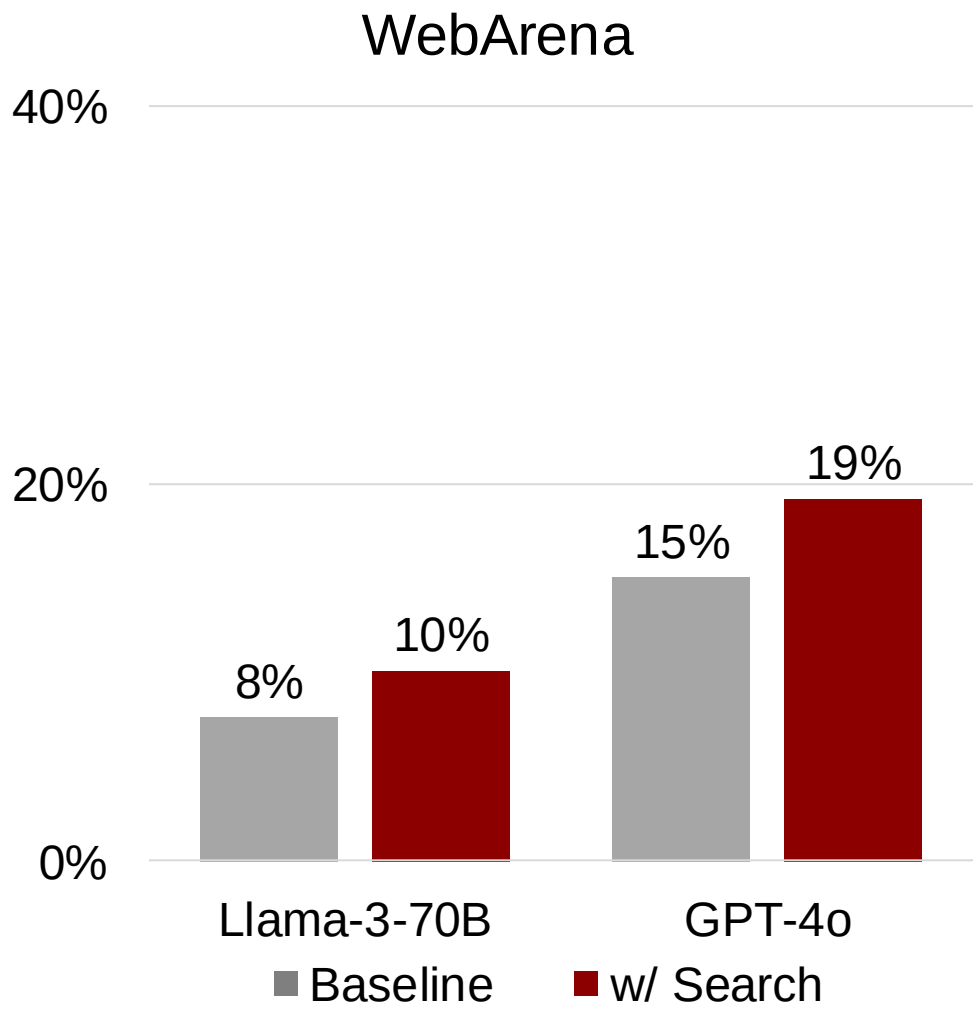
v = 1.0 State values

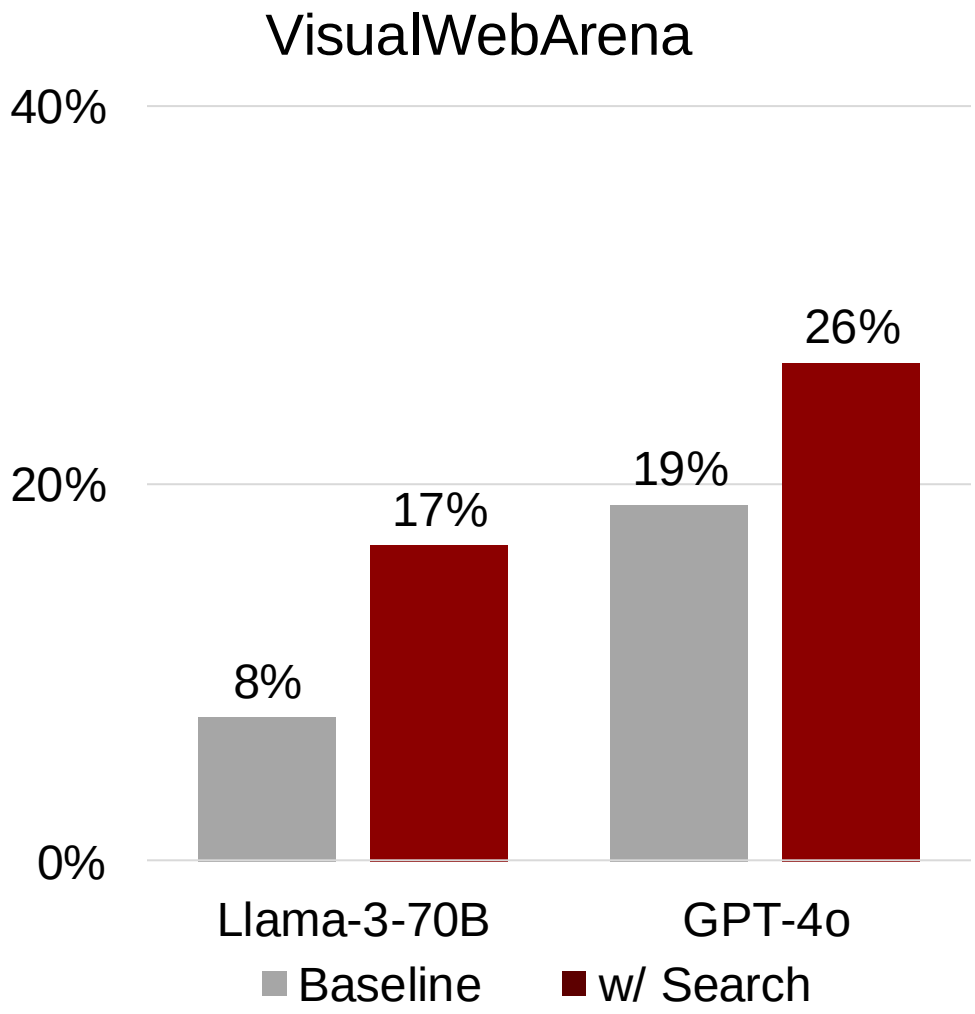
Backtracking

GPT-4o Agent

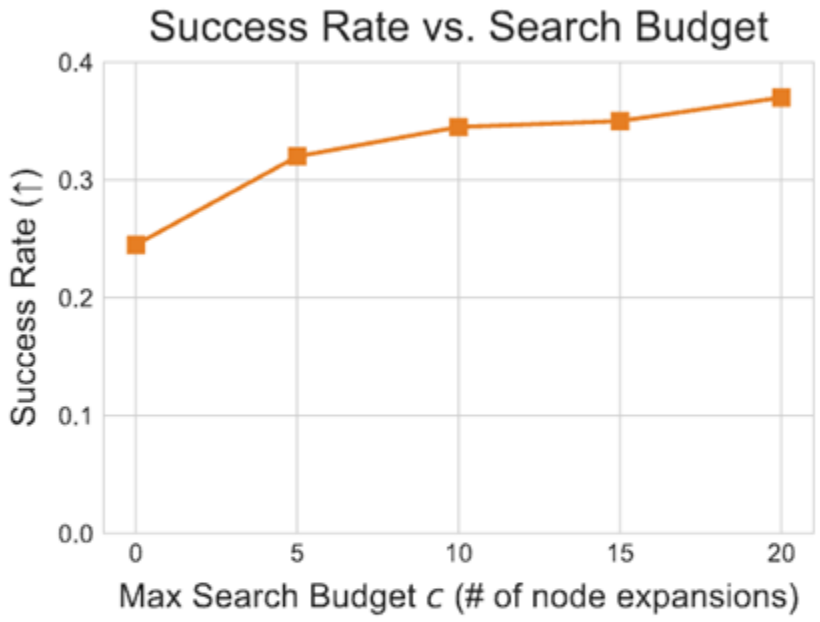
GPT-4o Agent + Search

Results


WebArena
40%
20%
0%
8%
10%
15%
19%
Llama-3-70B
GPT-4o
Baseline
w/ Search


VisualWebArena
40%
20%
0%
8%
17%
19%
26%
Llama-3-70B
GPT-4o
Baseline
w/ Search

Ablations



Success rate on a subset of 200 VWA tasks with search budget c. $c = 0$ indicates no search is performed. Success rate generally increases as c increases.

Depth d	**Branch** b	**SR** (↑)	Δ
0	1	24.5%	0%
1	3	26.0%	+6%
	5	32.0%	+31%
2	3	31.5%	+29%
	5	35.0%	+43%
3	5	35.5%	+45%
5	5	**37.0%**	+51%

Success rate (SR) and relative change over the baseline (Δ) on a subset of 200 VWA tasks with varying search depth (d) and branching factor (b). $d = 0$ indicates no search is performed. All methods use a max search budget $c = 20$.

Ablations

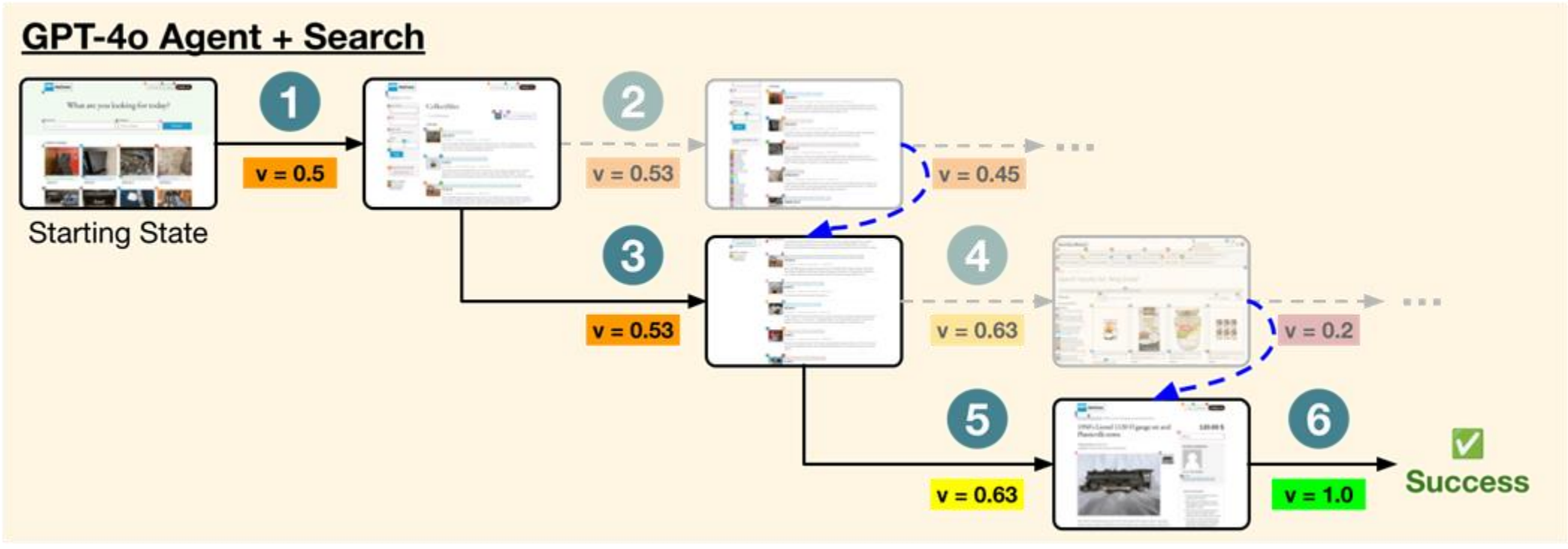
- Having a good value function is essential.
- There is still a lot of headroom for improving both the base agent policy, and the value function.

Value Function	SR (↑)
None (no search)	24.5%
LLaVA (w/ SC, $n = 20$)	30.0%
GPT-4o (no SC)	28.5%
GPT-4o (w/ SC, $n = 5$)	32.5%
GPT-4o (w/ SC, $n = 20$)	37.0%

Table 3: Success rate of the GPT-4o agent with different value functions.

Qualitative Results

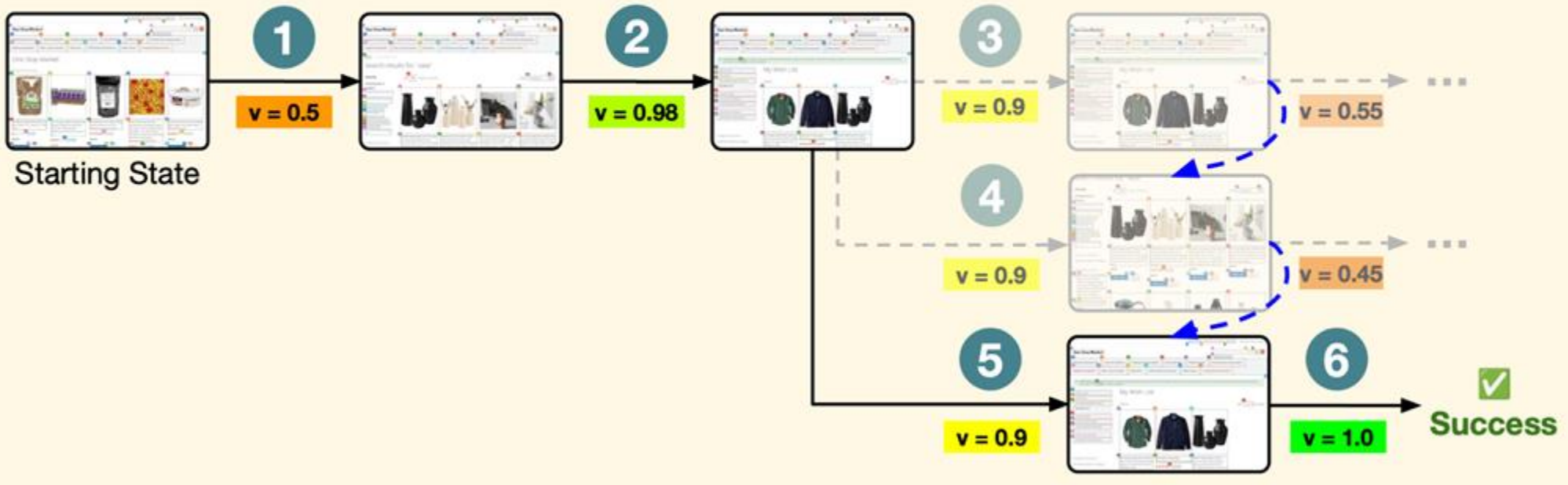
Task Instruction (I): "I recall seeing this exact item on the site, help me find the most recent post of it. I recall seeing it in either the Collectibles or Antiques section."



Qualitative Results

Task Instruction (I): "I need something like this for my apartment. Can you add one to my wishlist?"

GPT-4o Agent + Search



Legend: 1 Search sequence → Backtracking v = 1.0 State values

Limitations

- Search is slow
  - We implemented backtracking in a relatively naive way (store actions in a queue, take them again to get to the original state)

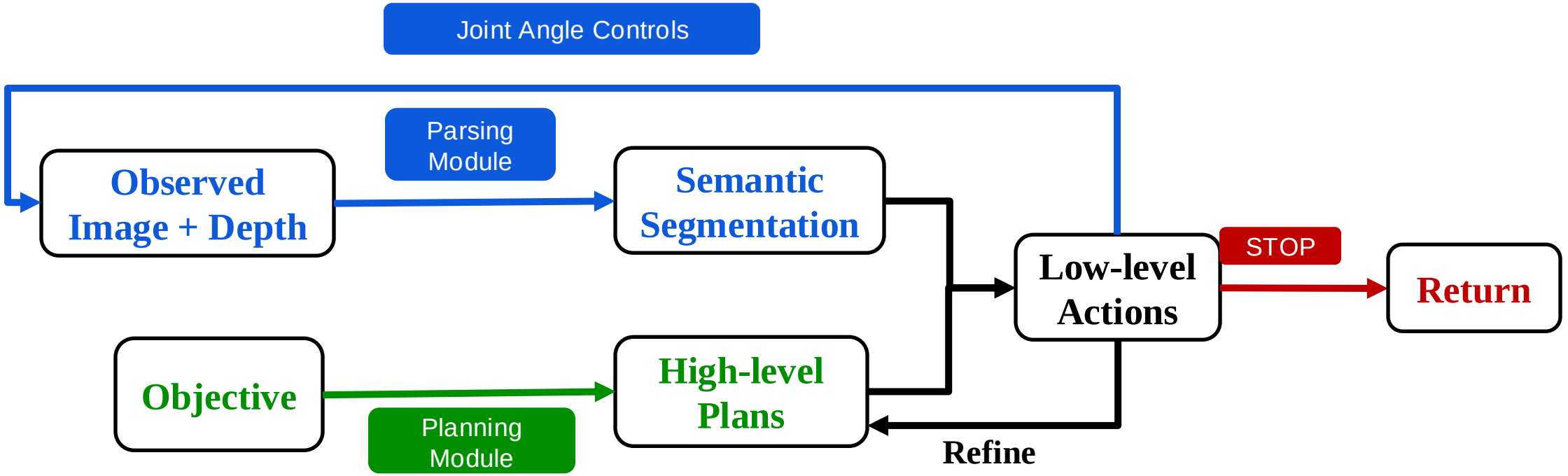
- Dealing with destructive actions
  - Some things on the web are very difficult to undo, e.g., ordering an item

Current Work

- Search as a policy improvement function
- Improving Value Function by fine-tuning instead of prompting
- Explore compute tradeoff between improving baseline agent vs. doing **more search at inference time**
- What if we don't have a perfect simulator – **how can we collect data at scale**?

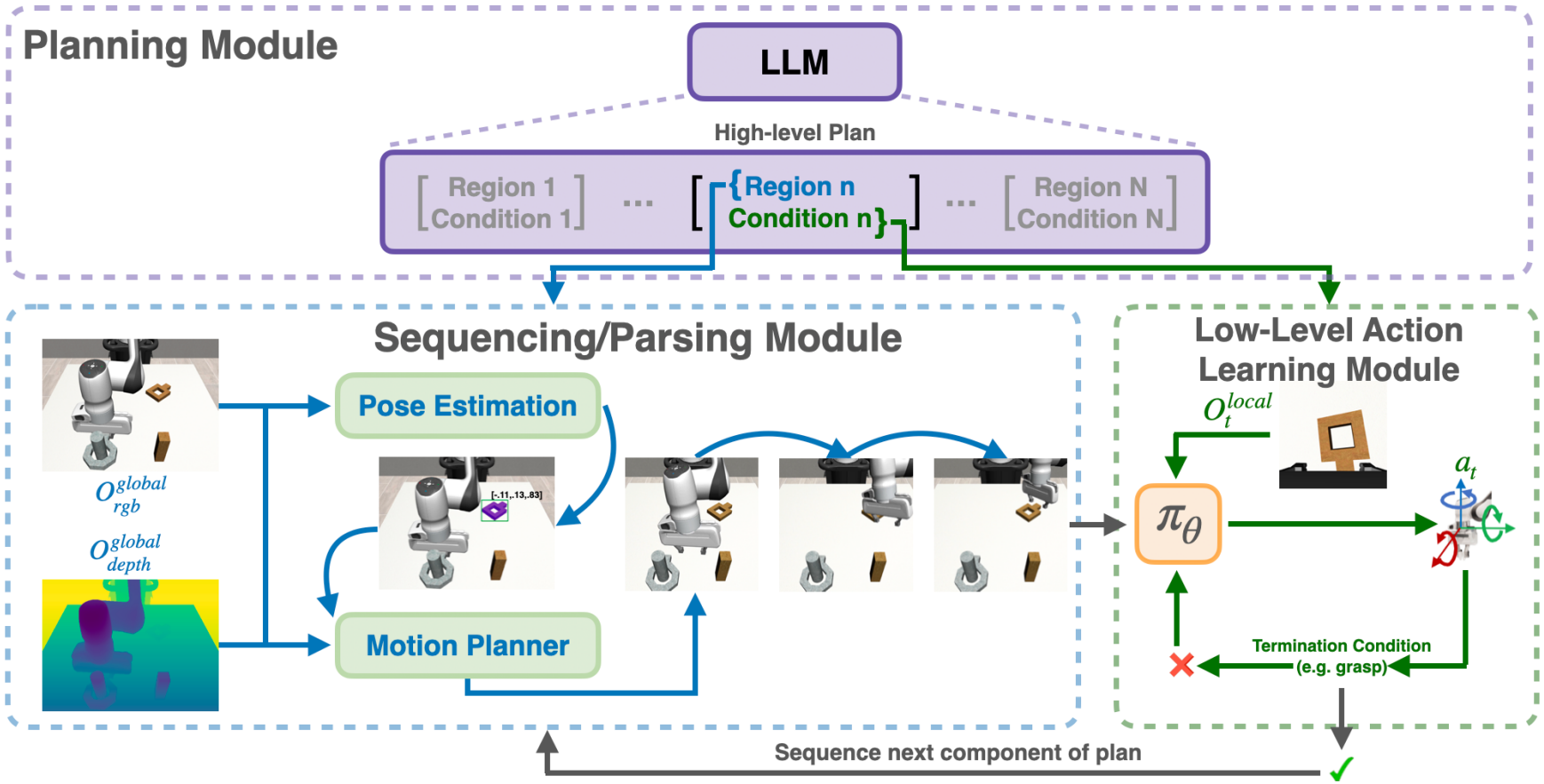
Physical Agent: Long-horizon Robotic Manipulation Task

- Model architecture of our interactive agent:
  - High-level Planning
  - Observation Parsing
  - Low-level Action Generation



Plan-Sequence-Learn

Murtaza Dalal



Plan-Seq-Learn (PSL): Language Model Guided RL for Solving Long Horizon Robotics, M Dalal, T Chiruvolu, D Chaplot, R Salakhutdinov, ICLR 2024

Planning Module

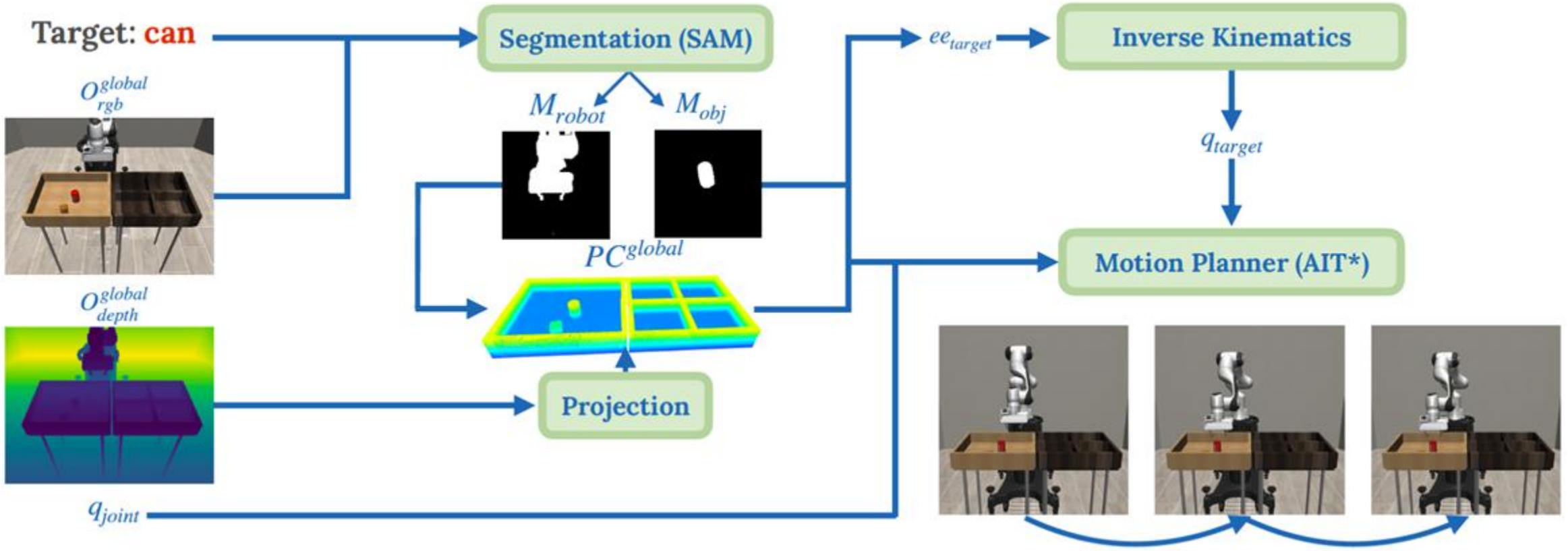
- Structured language plans: (object, condition)
- Prompt: Task description, conditions, objects, formatting

Stage termination conditions: (grasp, place).
Task description: The silver nut goes on the silver peg and the gold nut goes on the gold peg. Give me a simple plan to solve the task using only the stage termination conditions. Make sure the plan follows the formatting specified below and make sure to take into account object geometry.
Formatting of output: a list in which each element looks like: (<object/region>, <stage termination condition>). Don't output anything else.

Output:

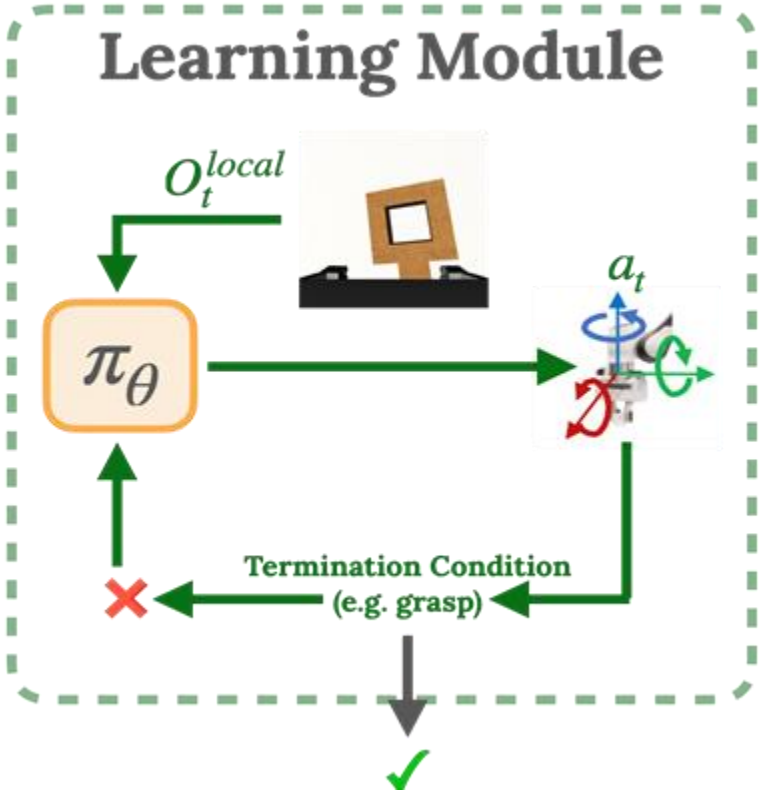
[(“silver nut”, “grasp”), (“silver peg”, “place”), (“gold nut”, “grasp”), (“gold peg”, “place”)]

Sequencing/Parsing Module: Grounding Language Plans in the Scene



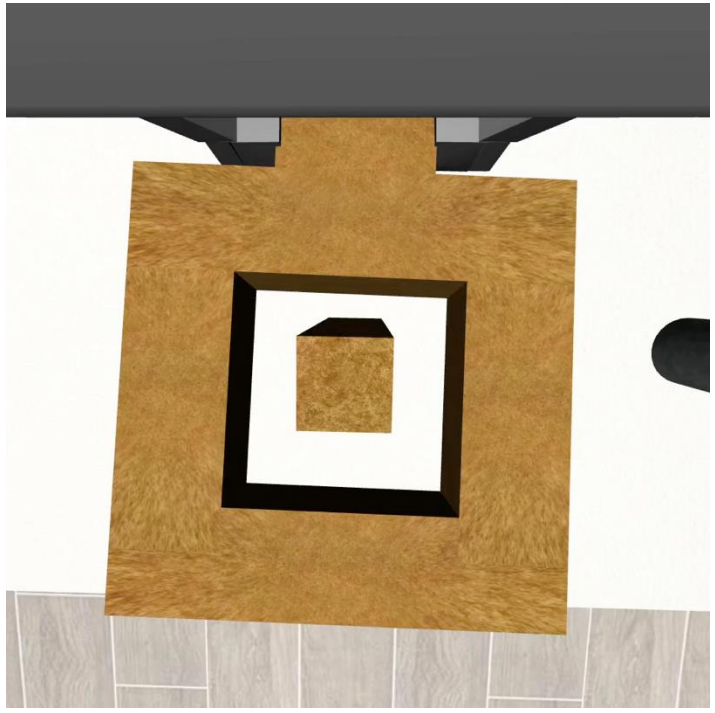
Learning Low-level Actions Module: Learning Local Control

- Learned RL policies for interaction
- Trained with task reward
- Single RL model instead of separate per stage
- Local instead of global observations

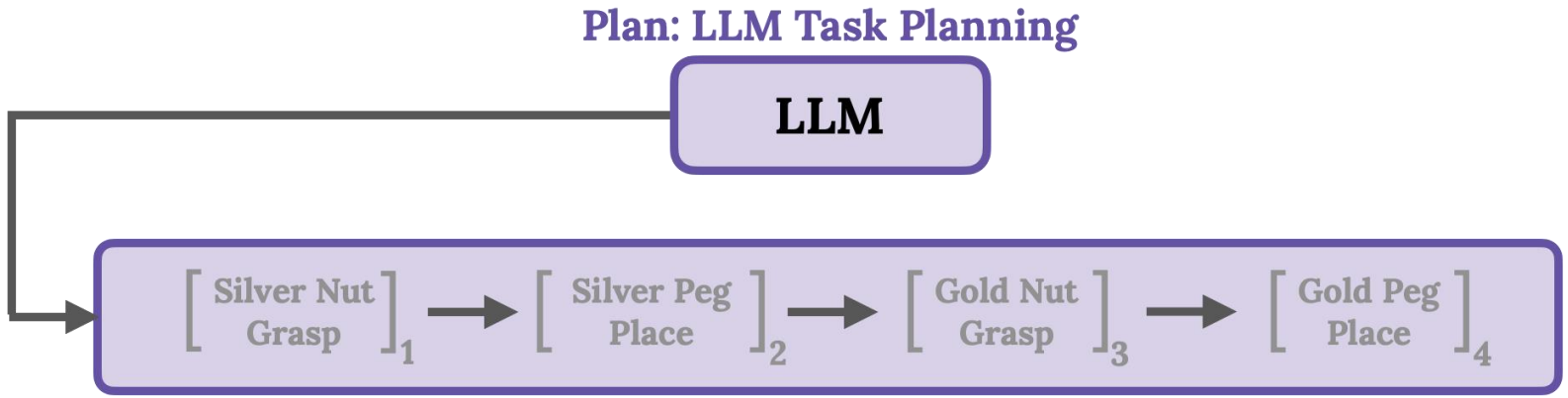


Learning Low-level Actions Module: Learning Local Control

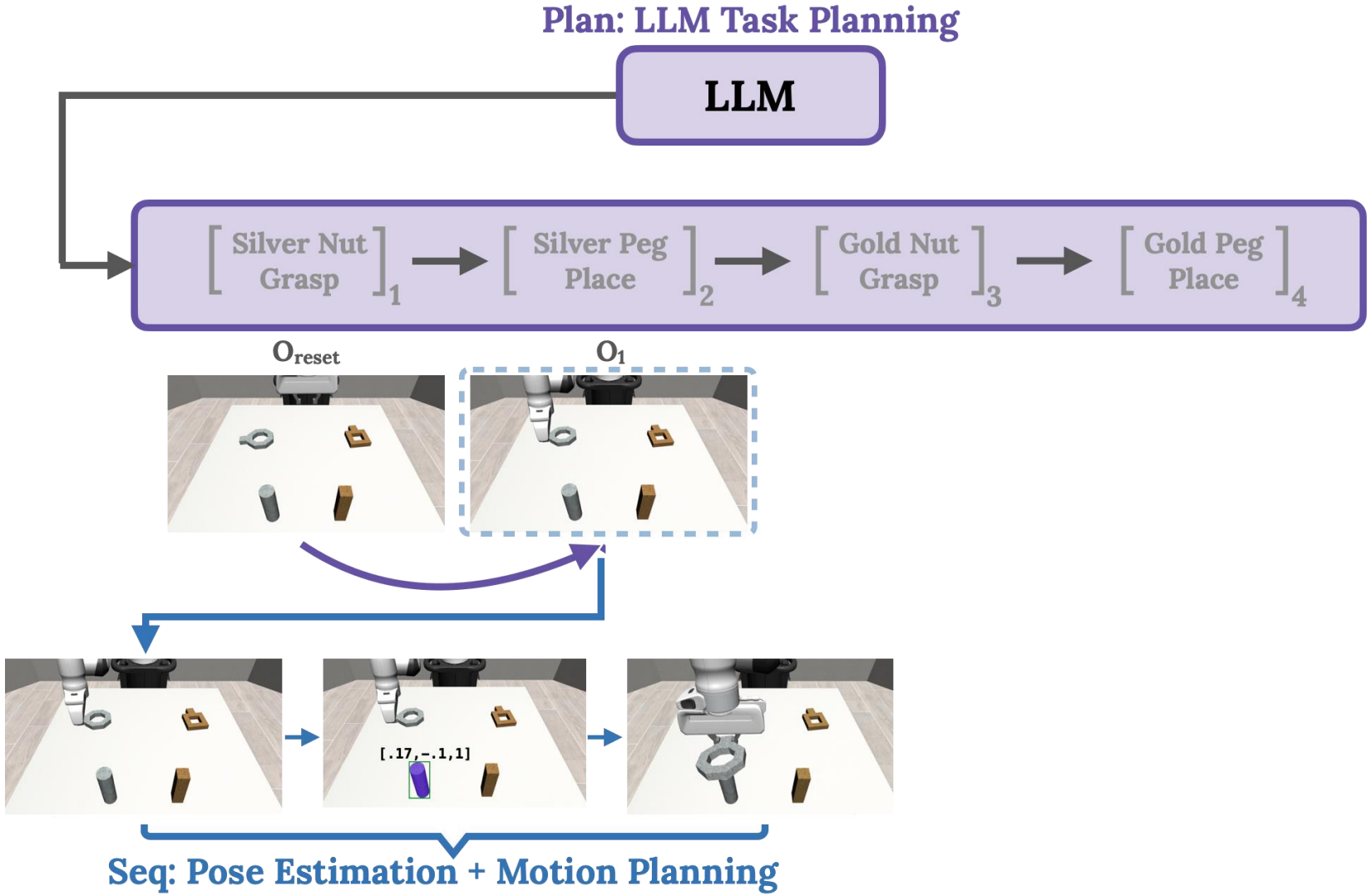
- Learned RL policies for interaction
- Trained with task reward
- Single RL model instead of separate per stage
- Local instead of global observations



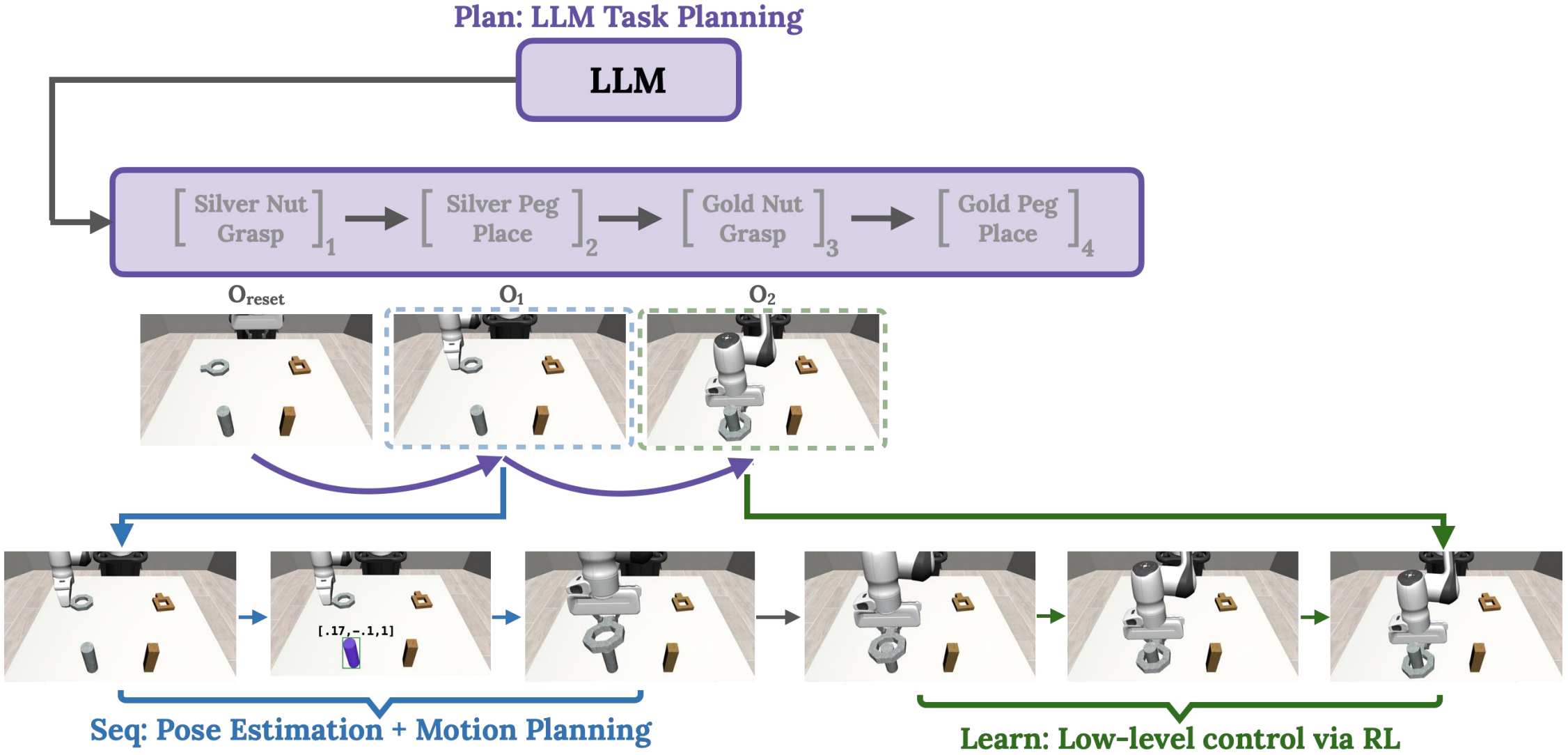
Full Pipeline Example



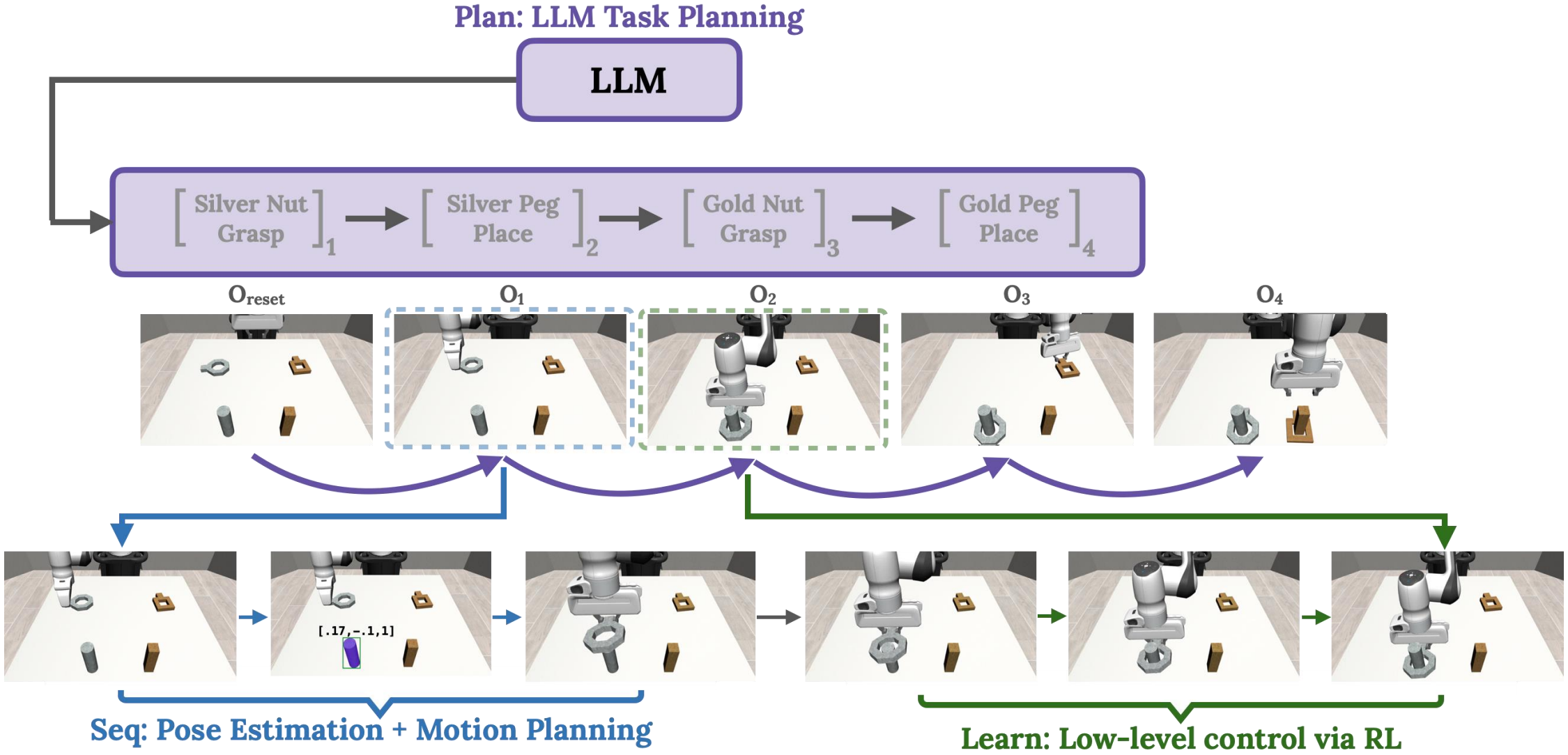
Full Pipeline Example

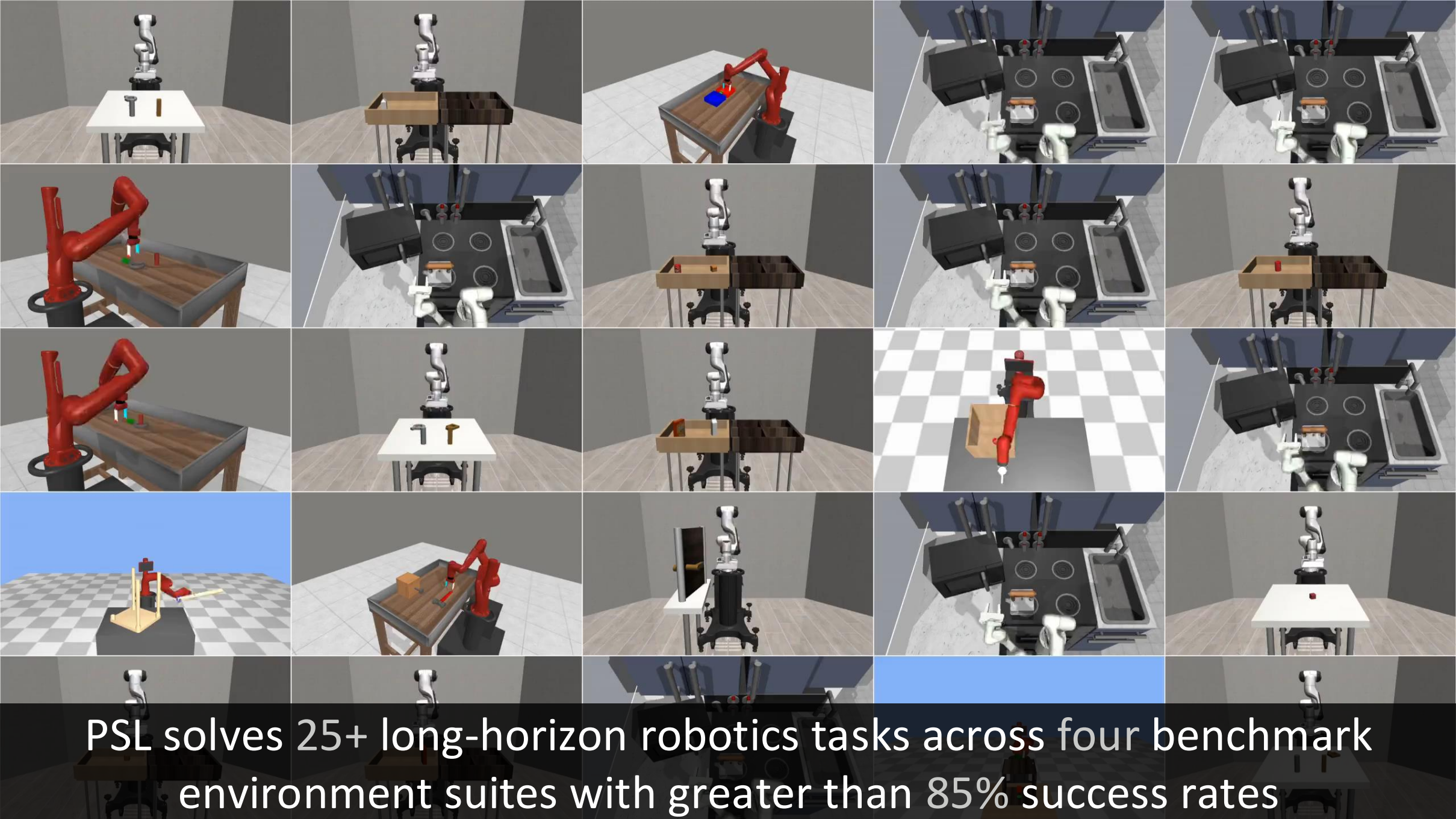


Full Pipeline Example

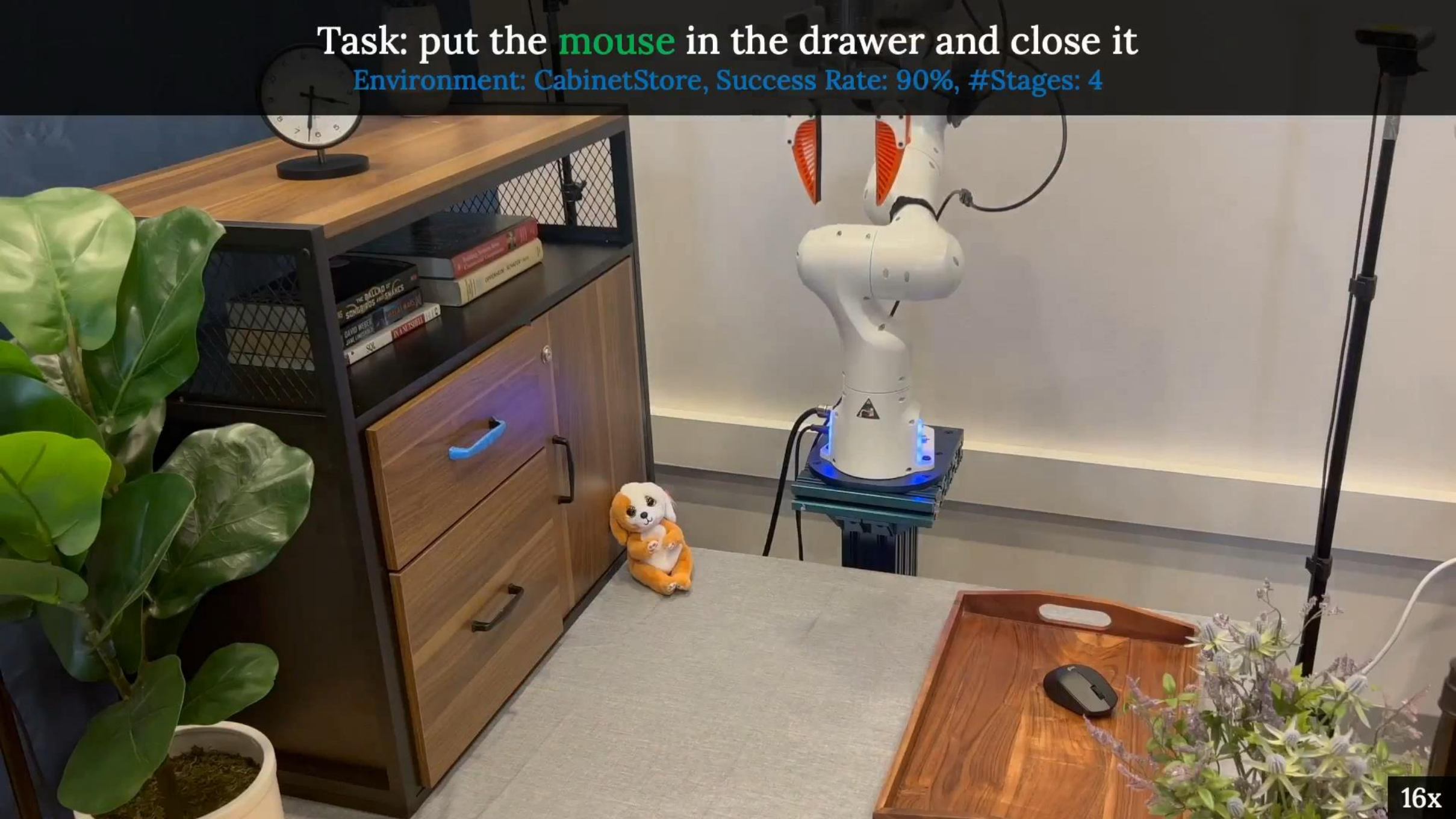


Full Pipeline Example

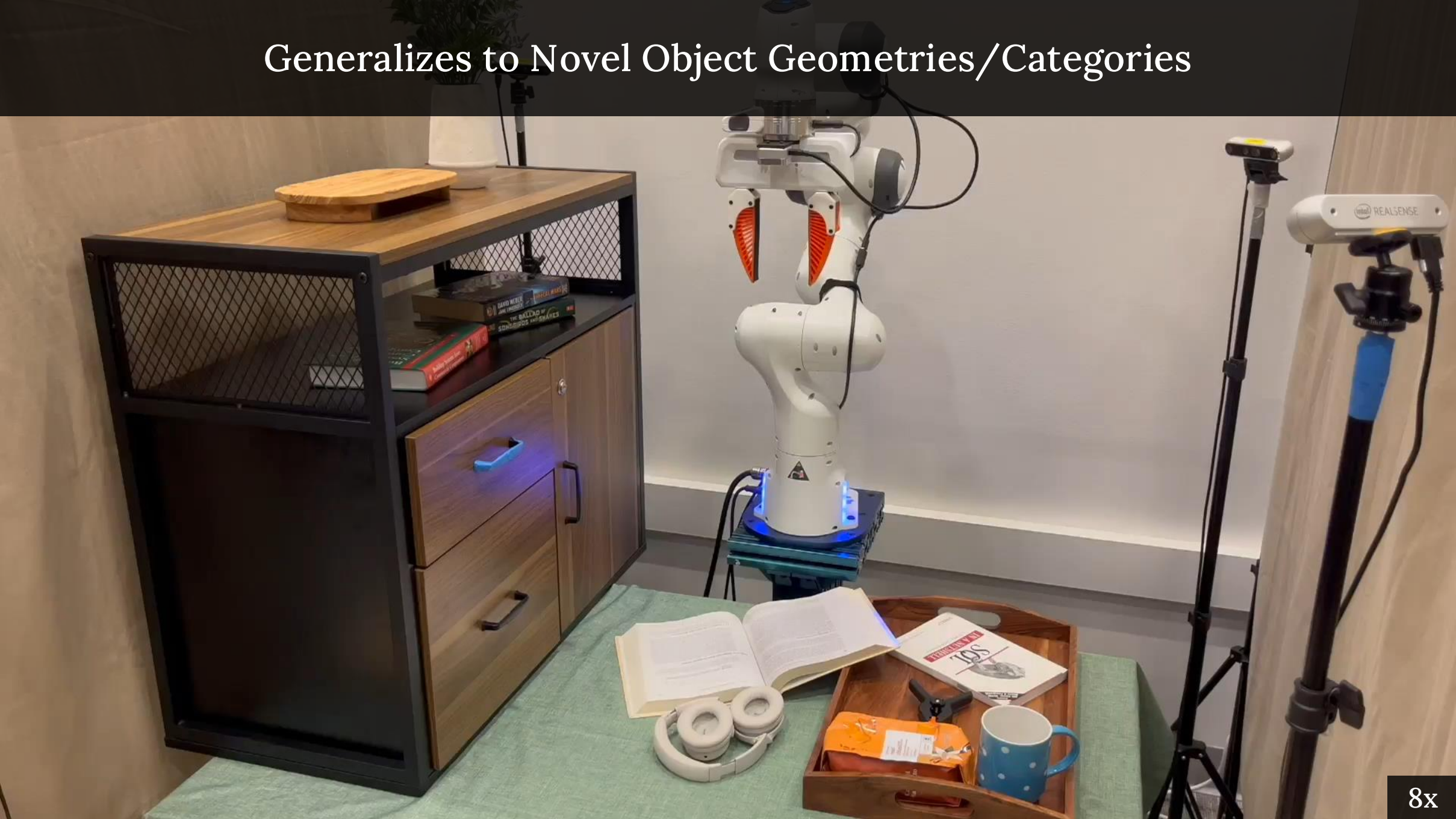




PSL solves 25+ long-horizon robotics tasks across four benchmark environment suites with greater than 85% success rates


Task: put the mouse in the drawer and close it
Environment: CabinetStore, Success Rate: 90%, #Stages: 4
16x

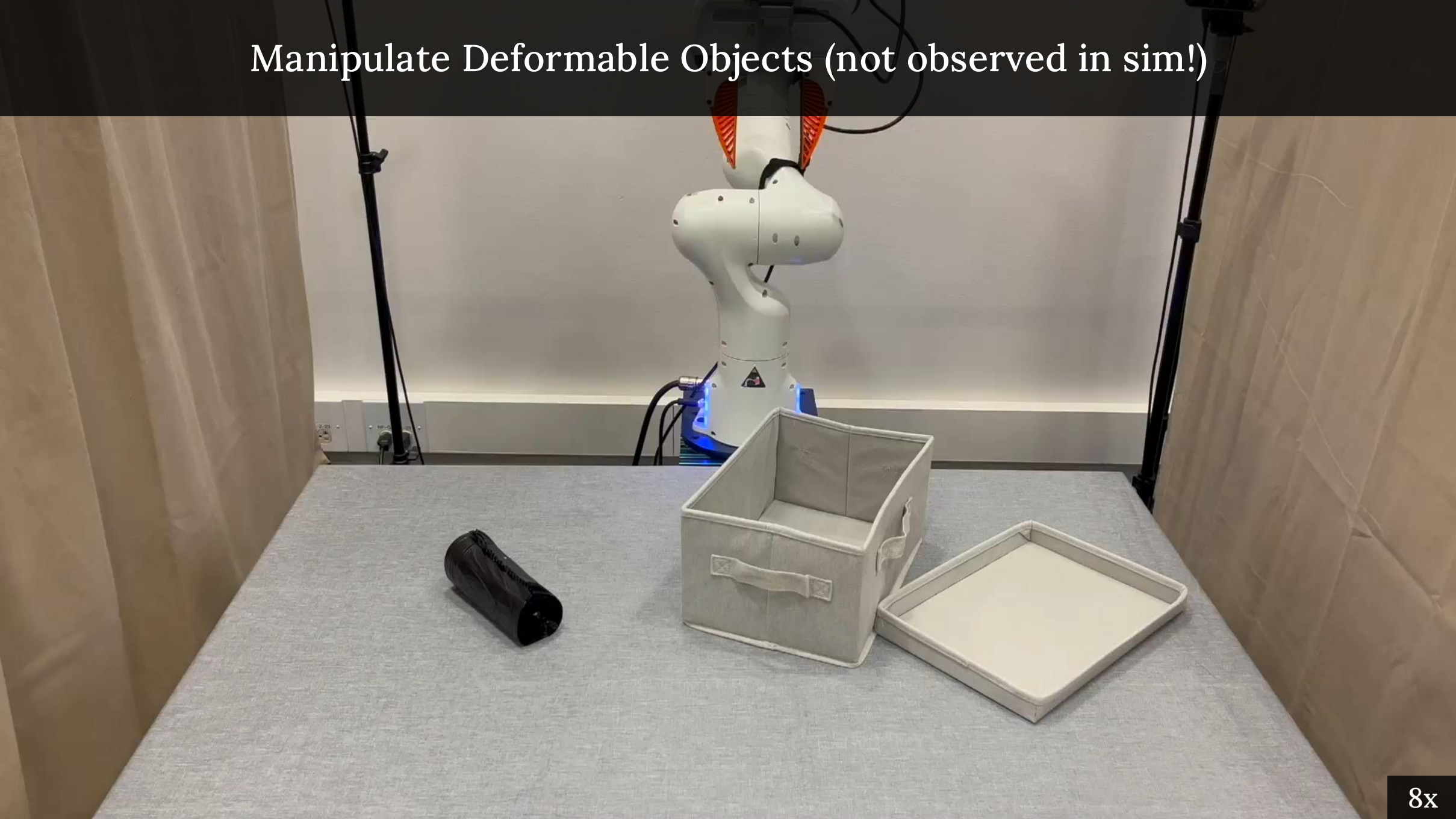
Generalizes to Novel Object Geometries/Categories



Manipulate novel objects with unseen receptacles

8x

Manipulate Deformable Objects (not observed in sim!)



8x

Summary

- VisualWebArena: a benchmark of realistic tasks designed to rigorously evaluate and advance the capabilities of autonomous multimodal web agents
- Inference-time search algorithm designed to enhance the capabilities of language model agents on realistic web tasks
- **AI Safety and robustness, especially in the age of autonomous systems.**

Adversarial Attacks on Multimodal Agents

Chen Henry Wu, Jing Yu Koh, Ruslan Salakhutdinov, Daniel Fried, Aditi Raghunathan
Carnegie Mellon University
{chenwu2,jingyuk,rsalakhu,dfried,aditirag}@cs.cmu.edu