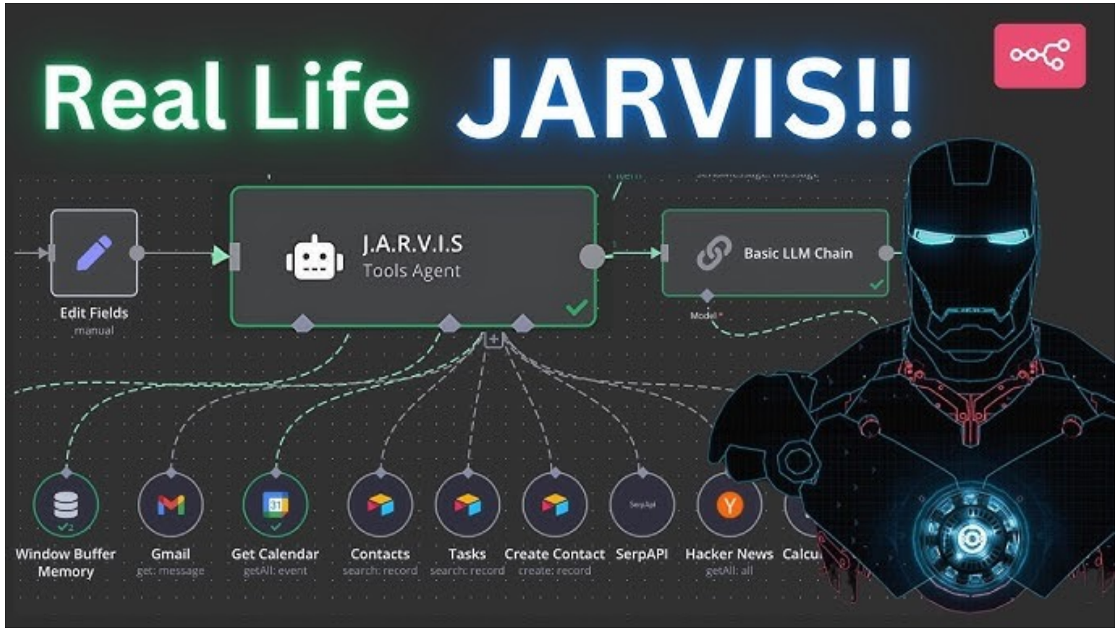
SCHOOL OF
COMPUTING &
DATA SCIENCE
The University of Hong Kong

Towards Generalist Computer-using Agents:

Models, Data, and Beyond

Qiushi Sun
qiushisun.github.io
𝕏 @qiushi_sun

Computer-using Agents



The Feasibility of Jarvis AI from Marvel in Real Life

[1] OS-Copilot: Towards Generalist Computer Agents with Self-Improvement

Computer-Using Agents

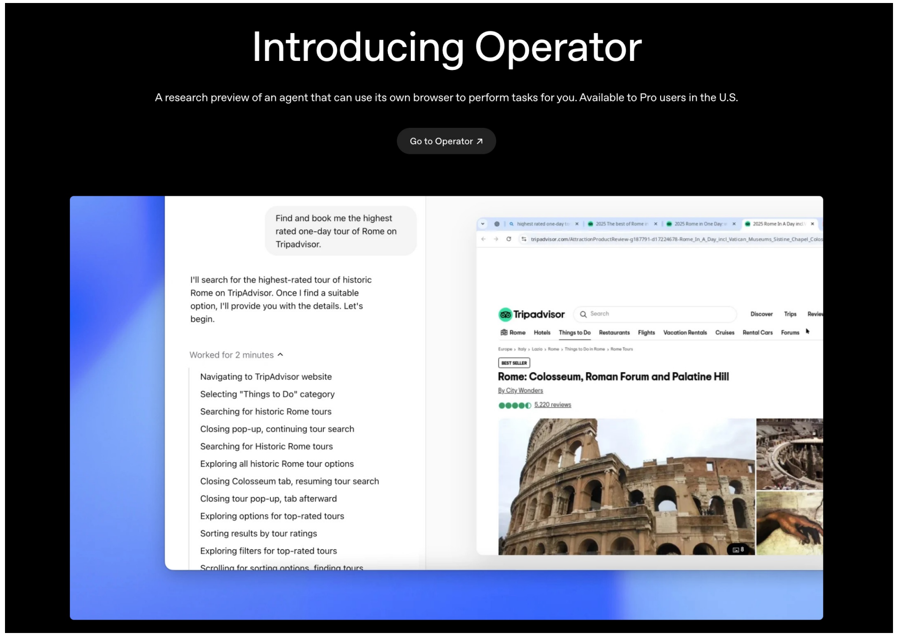
Both academia and industry are building computer-using agents

Claude Computer Use

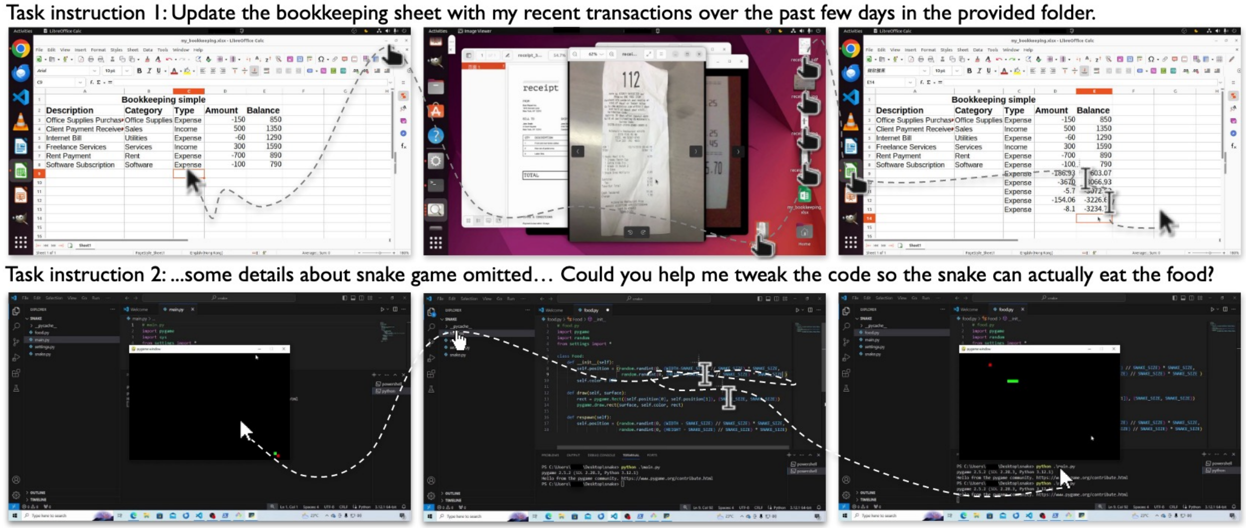
[2] *Introducing computer use, a new Claude 3.5 Sonnet, and Claude 3.5 Haiku, 22 Oct 2024*

Computer-Using Agents

Automating daily computer tasks



OpenAI Operator



Daily Computer Use

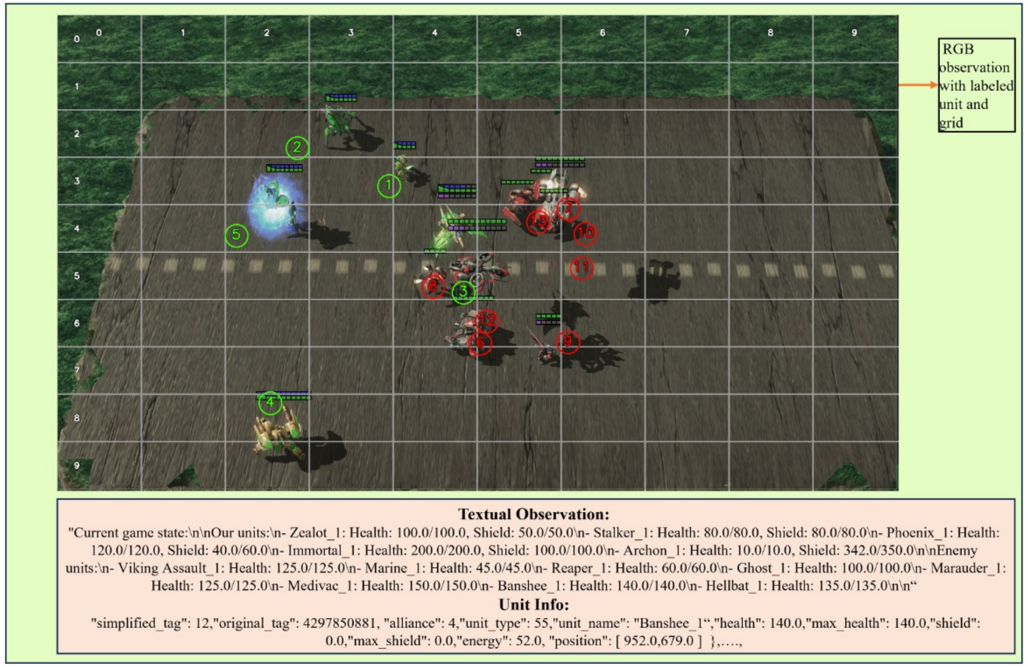
[3] *Introducing Operator: A research preview of an agent that can use its own browser to perform tasks for you., Jan 23, 2025*

[4] *OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments*

Computer-Using Agents

Playing Games

MineCraft

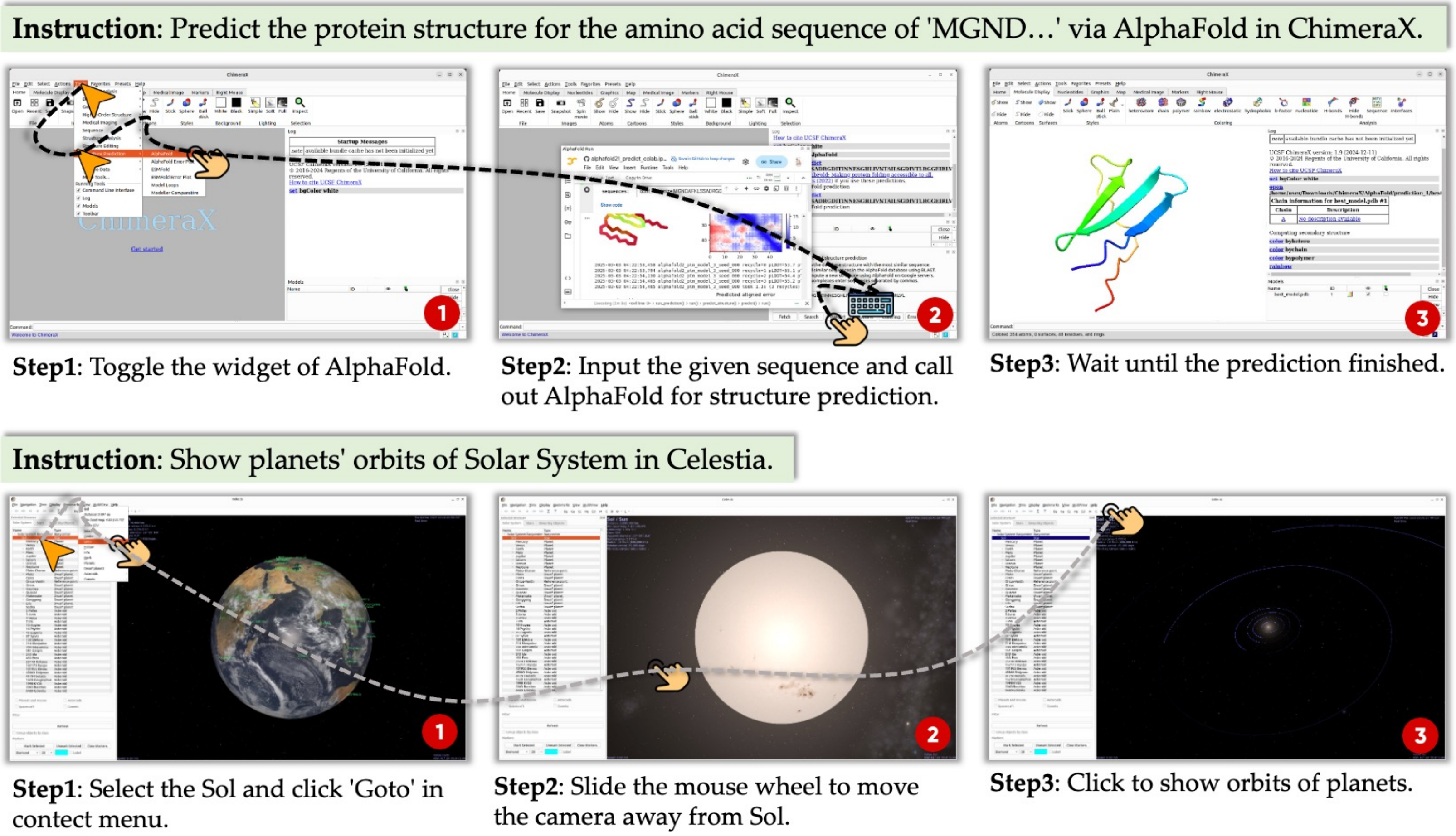


StarCraft II

[5] AVA: Attentive VLM Agent for Mastering StarCraft II

Computer-using Agents

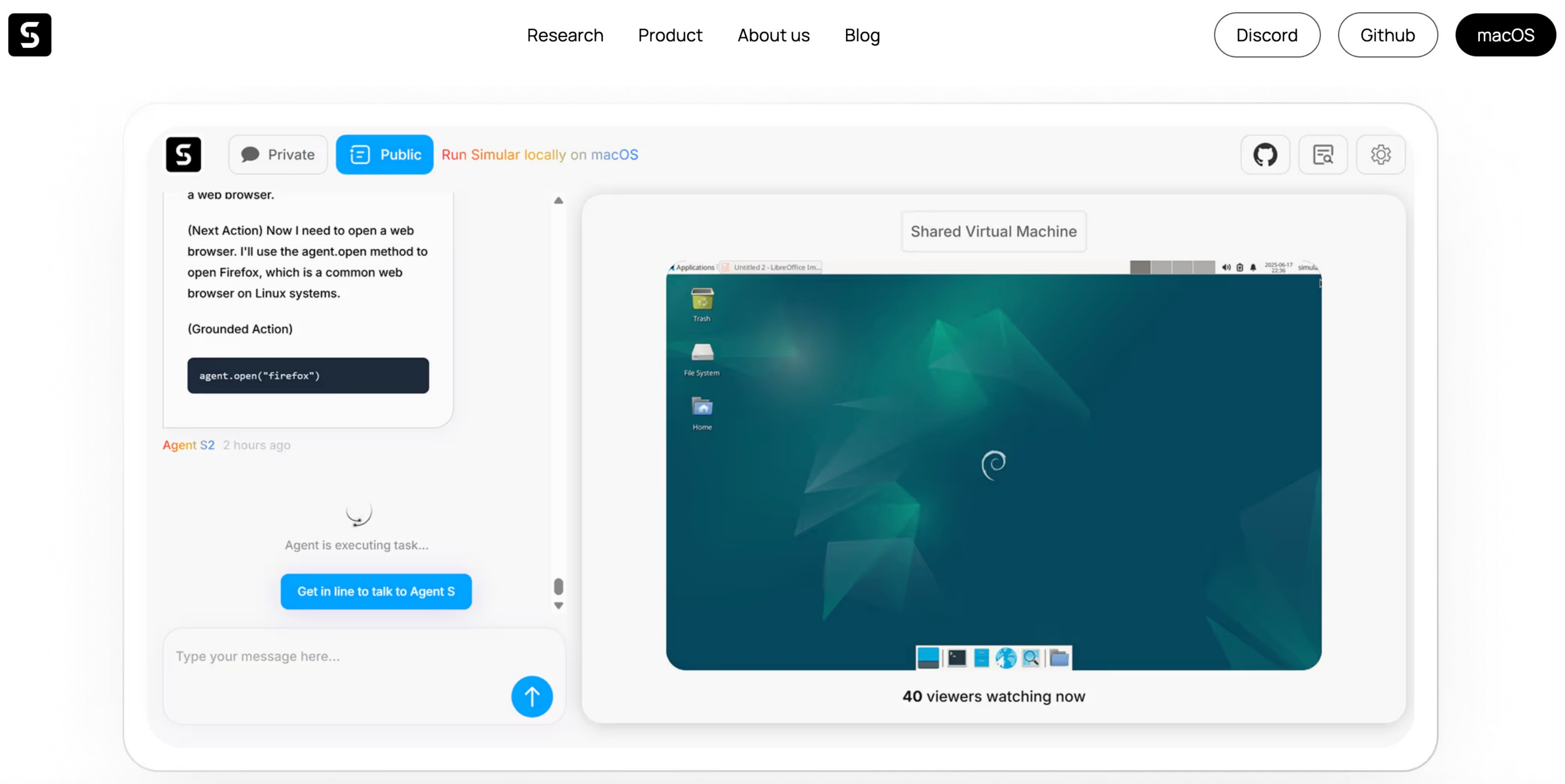
Automate scientific workflows, be your co-scientist



[6] ScienceBoard: Evaluating Multimodal Autonomous Agents in Realistic Scientific Workflows

Computer-using Agents

Startups



Seminal works on Computer-Using Agents

Foundation Models

SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents, ACL 2024

OS-ATLAS: A Foundation Action Model for Generalist GUI Agents , ICLR 2025 Spotlight

Data

OS-Genesis: Automating GUI Agent Trajectory Construction via Reverse Task Synthesis , ACL 2025

Breaking the Data Barrier -- Building GUI Agents Through Task Generalization, COLM 2025

Algorithm

AgentStore: Scalable Integration of Heterogeneous Agents As Specialized Generalist Computer Assistant , ACL 2025

Evaluation

OS-MAP: How Far Can Computer Use Agents Go in Breadth and Depth?

Frontier Application

ScienceBoard: Evaluating Multimodal Autonomous Agents in Realistic Scientific Workflows

Safety

OS-Sentinel : Towards Safety-Enhanced Mobile GUI Agents via Hybrid Validation in Realistic Workflows

Computer-Using Agents

Generally, both GUI and CLI can enable computer use

(though they have different capability boundaries).

Today, our discussion focuses on GUI-based computer-using agents.

GUI Agents

SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents

ACL 2024
Bangkok, Thailand

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, Zhiyong Wu

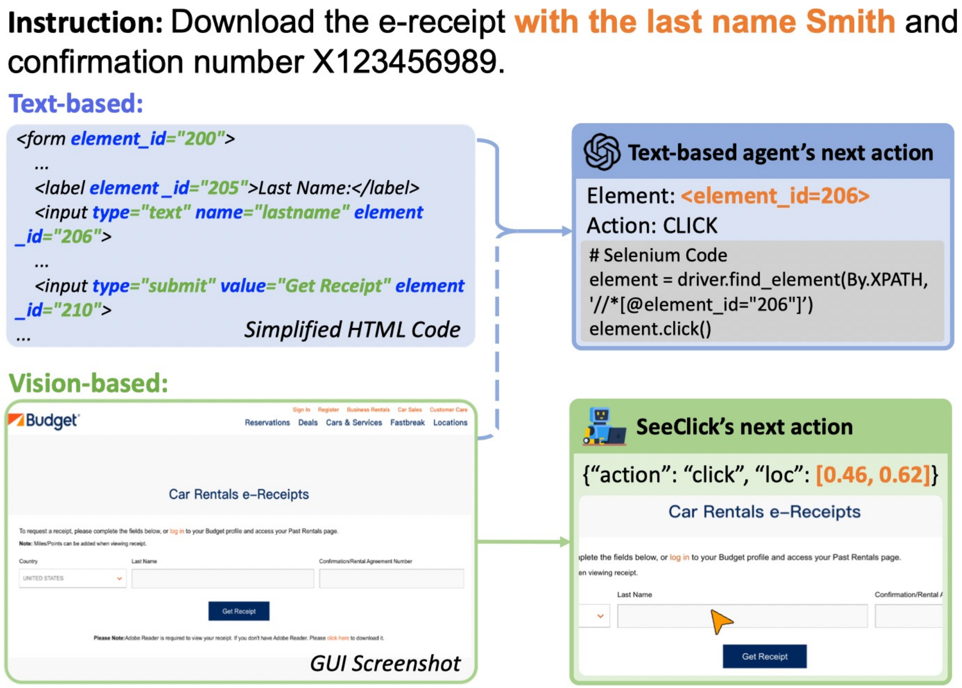
上 海 人 工 智 能 实 验 室
Shanghai Artificial Intelligence Laboratory

NANJING UNIVERSITY 1902

SeeClick: Overview

We built a purely visual GUI Agent SeeClick, which interacts with GUIs through screenshots, does not require any structured information.

Just like Human!



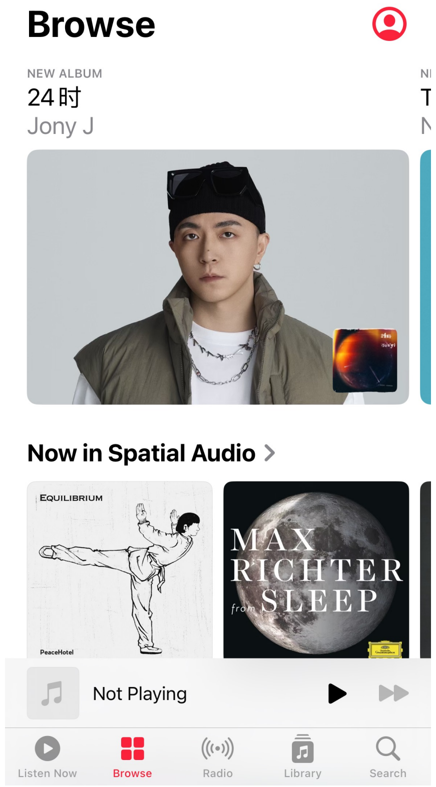
Input: Screenshots

Output: the action (with location)

[7] SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents, ACL 2024

SeeClick: GUI Grounding

We discovered a key challenge in developing visual GUI agents: GUI grounding – the capacity to accurately locate screen elements based on instructions.



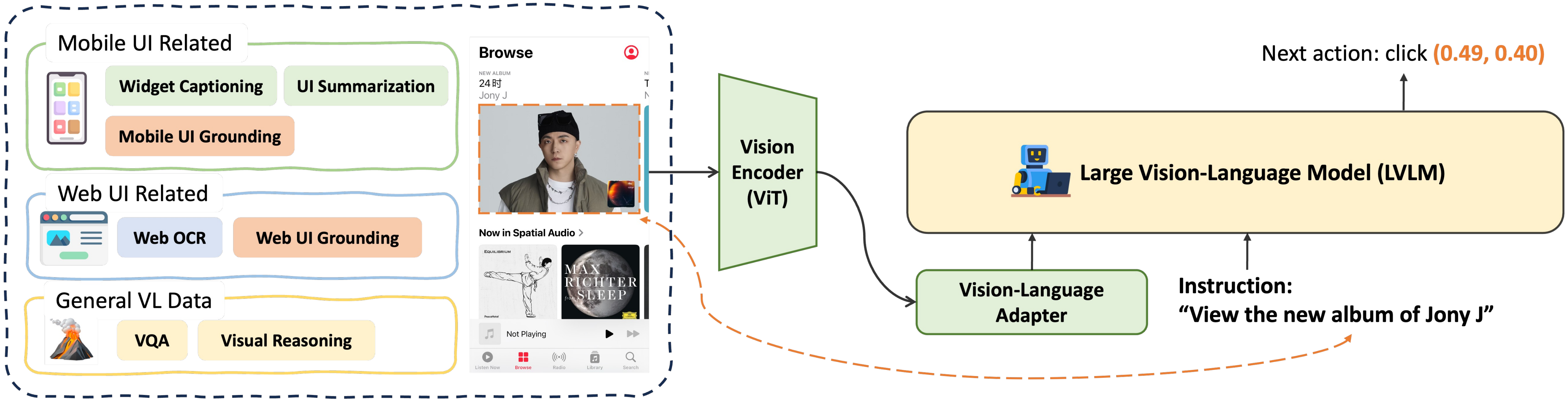
In order to view the new album of Jony J, where should I click?

GPT-4o (an earlier version): hmmm… Sorry I don’t know. ❌

SeeClick: (0.49, 0.40) ✅

How SeeClick is Built

Overview of SeeClick's framework and GUI grounding pre-training.



Uses ~1M GUI-specific samples combining web UI, mobile UI, and general vision-language data.

Includes **GUI grounding tasks**, such as predicting click points and generating element descriptions.

How SeeClick is Built

Web UI Grounding data

1. Crawled from large-scale web pages (~300K pages)
2. Includes text elements and tooltip-based descriptions

instructions

Target: element localization from instructions $p(y|s,x)$ and OCR-style text prediction $p(x|s,y)$

elements

Mobile UI data

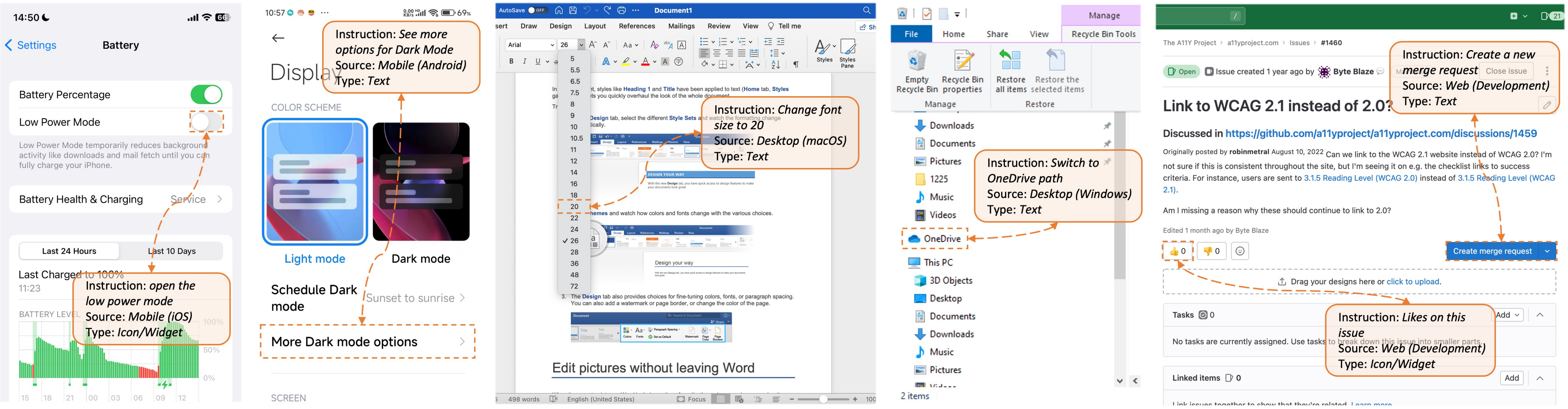
1. Widget captioning and UI grounding from public datasets (e.g., RICO)
2. UI summarization to improve holistic interface understanding

General VL instruction data

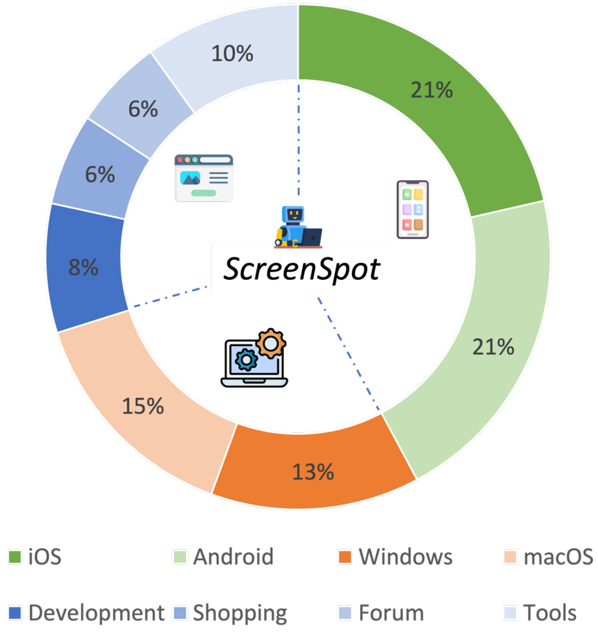
1. Adopted from multi-purpose VL instruction-following corpora (e.g., LLaVA)
2. Supports preserving general reasoning and descriptive capabilities

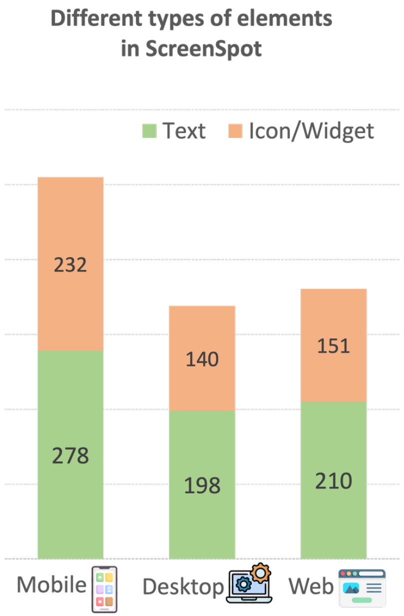
The First Modern GUI Grounding Benchmark

GUI Grounding Benchmark: *ScreenSpot*



The First Modern GUI Grounding Benchmark





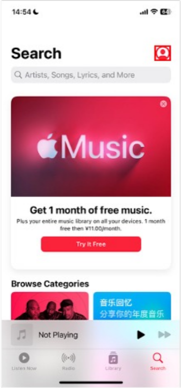
600+ screenshots and **1,200+ instructions** across mobile (iOS, Android), desktop (macOS, Windows), and web platforms.

Both **text elements and icons/widgets**

Collected from real-world apps and websites

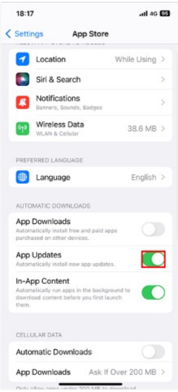
ScreenSpot: Component

Mobile

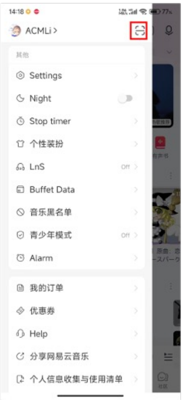


Instruction: *My account*
Source: *Mobile (iOS)*
Type: *Icon/Widget*

Instruction: *Remove maps from the Desktop*
Source: *Mobile (iOS)*
Type: *Icon/Widget*

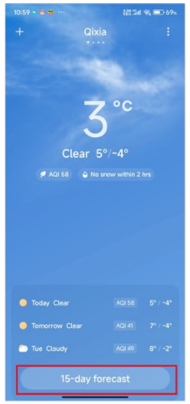


Instruction: *Disallow automatic app updates*
Source: *Mobile (iOS)*
Type: *Icon/Widget*

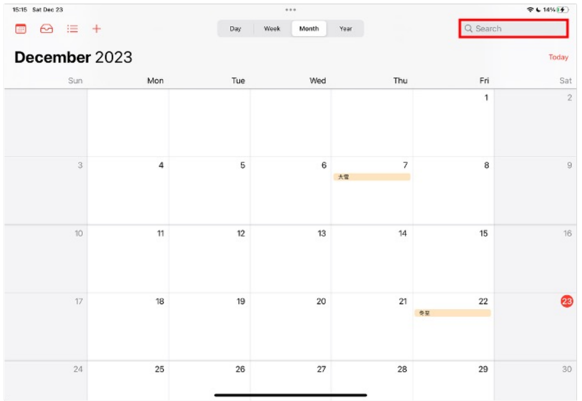


Instruction: *Scan QR code*
Source: *Mobile (Android)*
Type: *Icon/Widget*

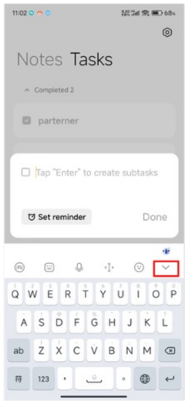
Instruction: *Continue*
Source: *Mobile (Android)*
Type: *Text*



Instruction: *Display 15-day weather forecast*
Source: *Mobile (Android)*
Type: *Text*



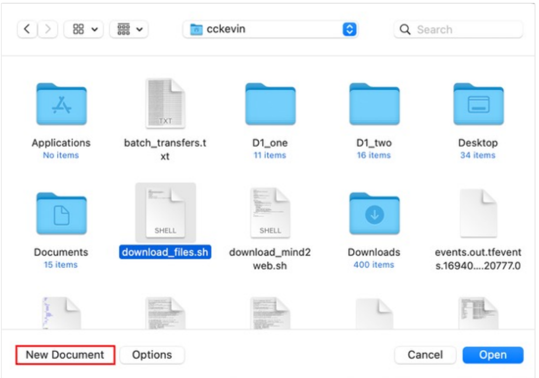
Instruction: *Search event*
Source: *Mobile (iOS)*
Type: *Text*



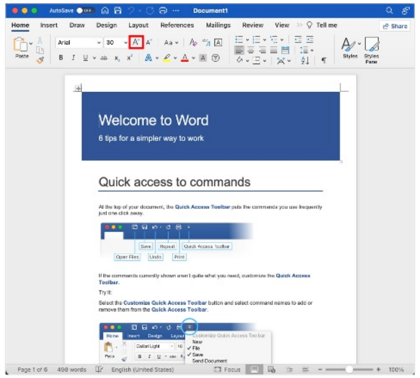
Instruction: *Fold input method*
Source: *Mobile (Android)*
Type: *Icon/Widget*

ScreenSpot: Component

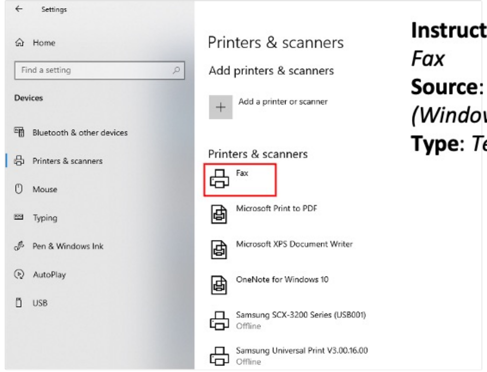
Desktop



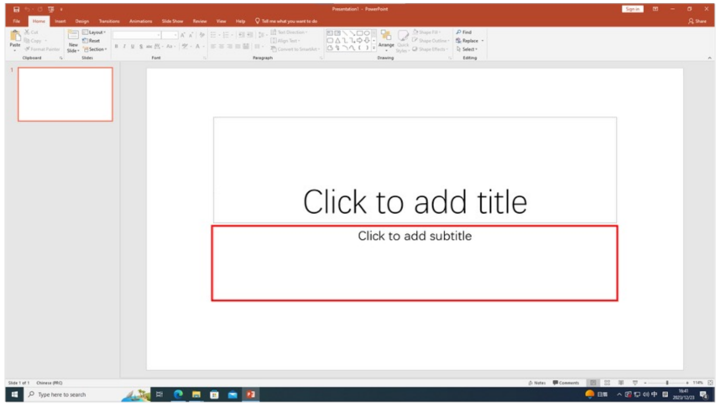
Instruction: *Create a new document*
Source: *Desktop (macOS)*
Type: *Text*



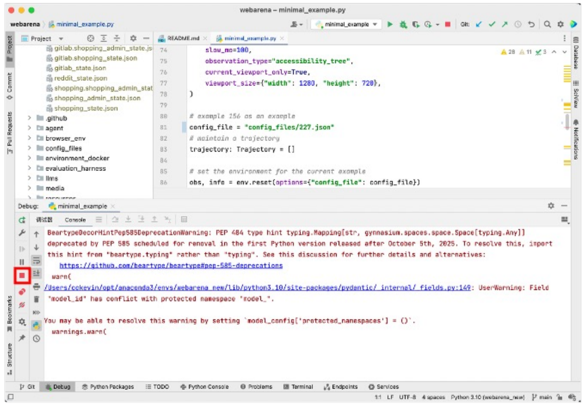
Instruction: *Enlarge font size*
Source: *Desktop (macOS)*
Type: *Icon/Widget*



Instruction: *Open Fax*
Source: *Desktop (Windows)*
Type: *Text*



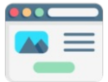
Instruction: *Add subtitle*
Source: *Desktop (Windows)*
Type: *Text*

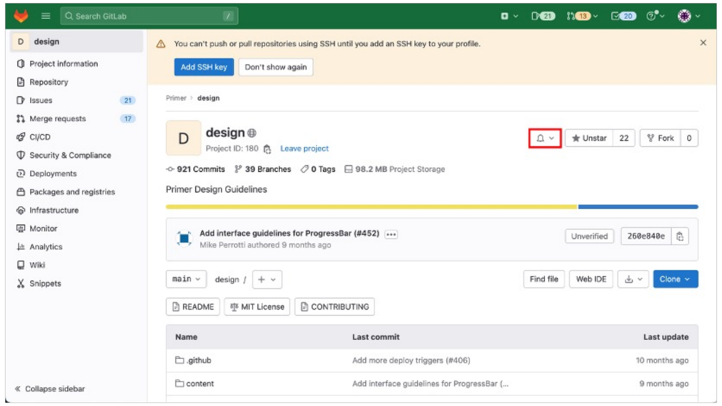


Instruction: *Pause the debugger*
Source: *Desktop (macOS)*
Type: *Icon/Widget*

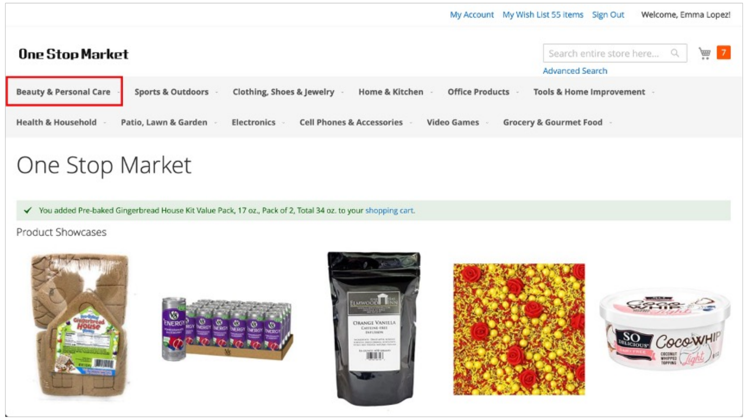
ScreenSpot: Component

Web

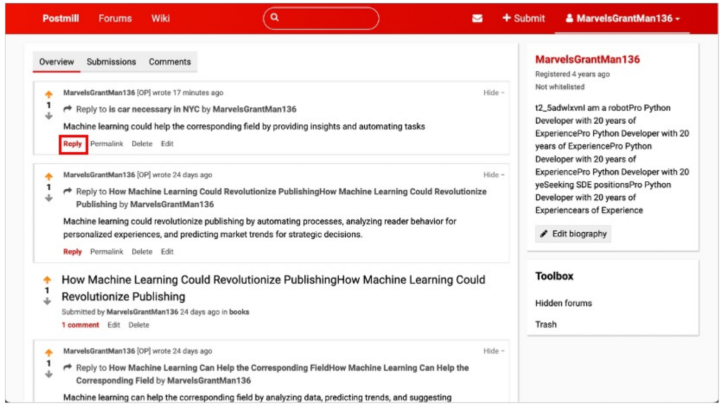




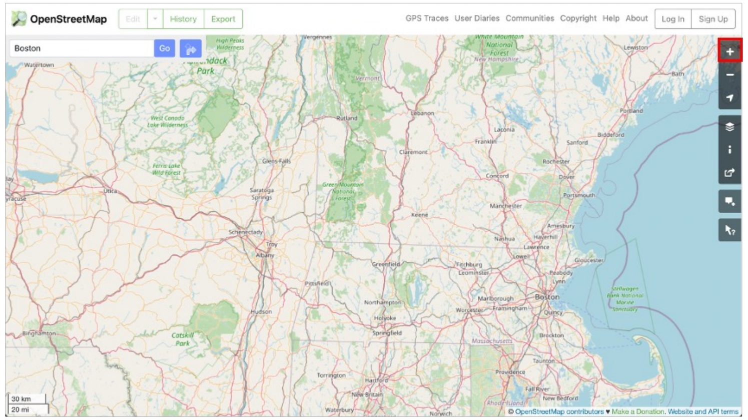
Instruction: *Set Reminder*
Source: *Web (Development)*
Type: *Icon/Widget*



Instruction: *Go to Beauty & Personal Care*
Source: *Web (Shop)*
Type: *Text*

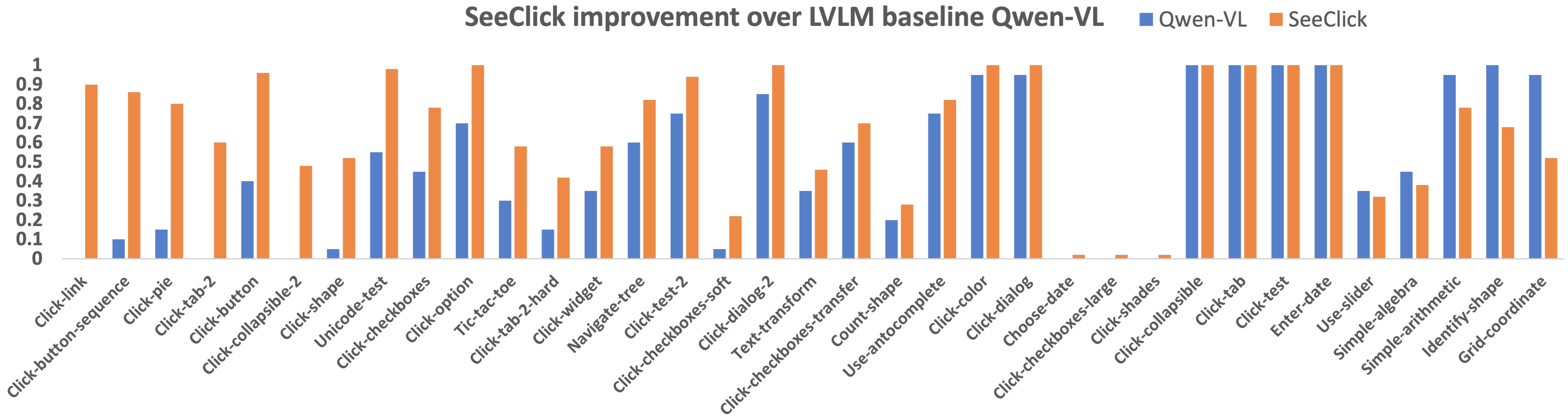


Instruction: *Reply to the first post*
Source: *Web (Forum)*
Type: *Text*



Instruction: Zoom in on the map
Source: *Web (Tools)*
Type: *Icon/Widget*

Results on ScreenSpot



ScreenSpot's Far-reaching Impact

ScreenSpot

ScreenSpot V2

A new ScreenSpot with error correction

ScreenSpot-Pro

Specifically designed for high-res professional scenarios (e.g., Photoshop, MATLAB), featuring more complex tasks.

ScreenSuite

Treat ScreenSpot as an evaluation component

MMBench-GUI

Treat ScreenSpot as an evaluation component

[8] OS-ATLAS: A Foundation Action Model For Generalist GUI Agents, ICLR 2025 Spotlight

[9] ScreenSpot-Pro: GUI Grounding for Professional High-Resolution Computer Use

[10] ScreenSuite - The most comprehensive evaluation suite for GUI Agents!

OS-ATLAS: A Foundation Action Model For Generalist GUI Agents

ICLR 2025 Spotlight

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, Qiao Yu

上海人工智能实验室
Shanghai Artificial Intelligence Laboratory

The Road of Building GUI Agent

Still, a vision-only solution

- Previous: html / a11ytree as states
- Trending: screenshots as states (human-like)

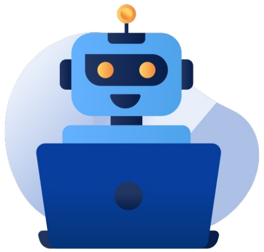
Importance of Large Action Model

Planner

+

Action Model

=

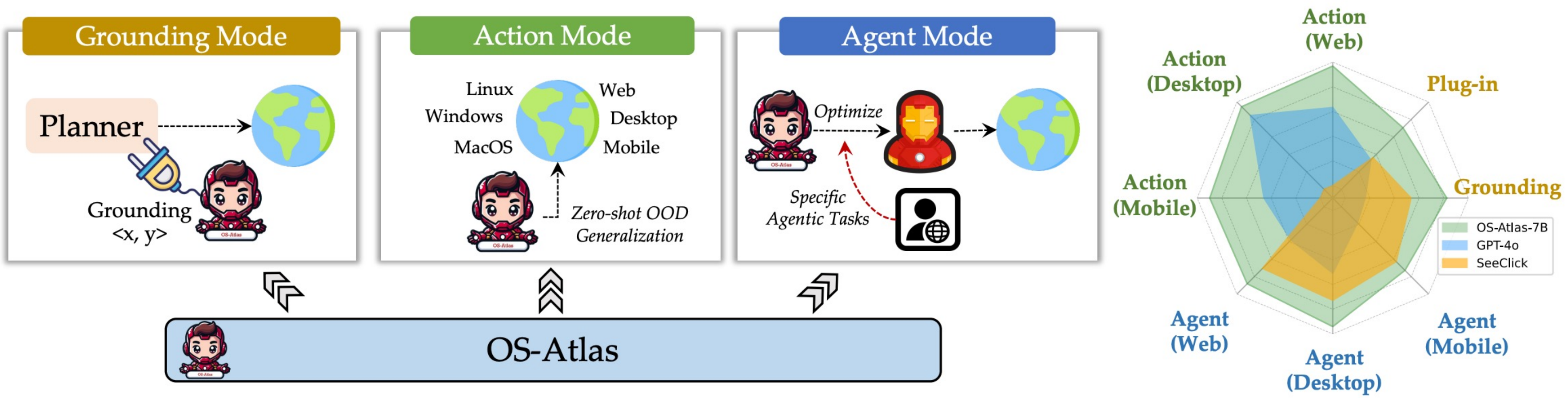


Minimal Agent

Overview of OS-Atlas

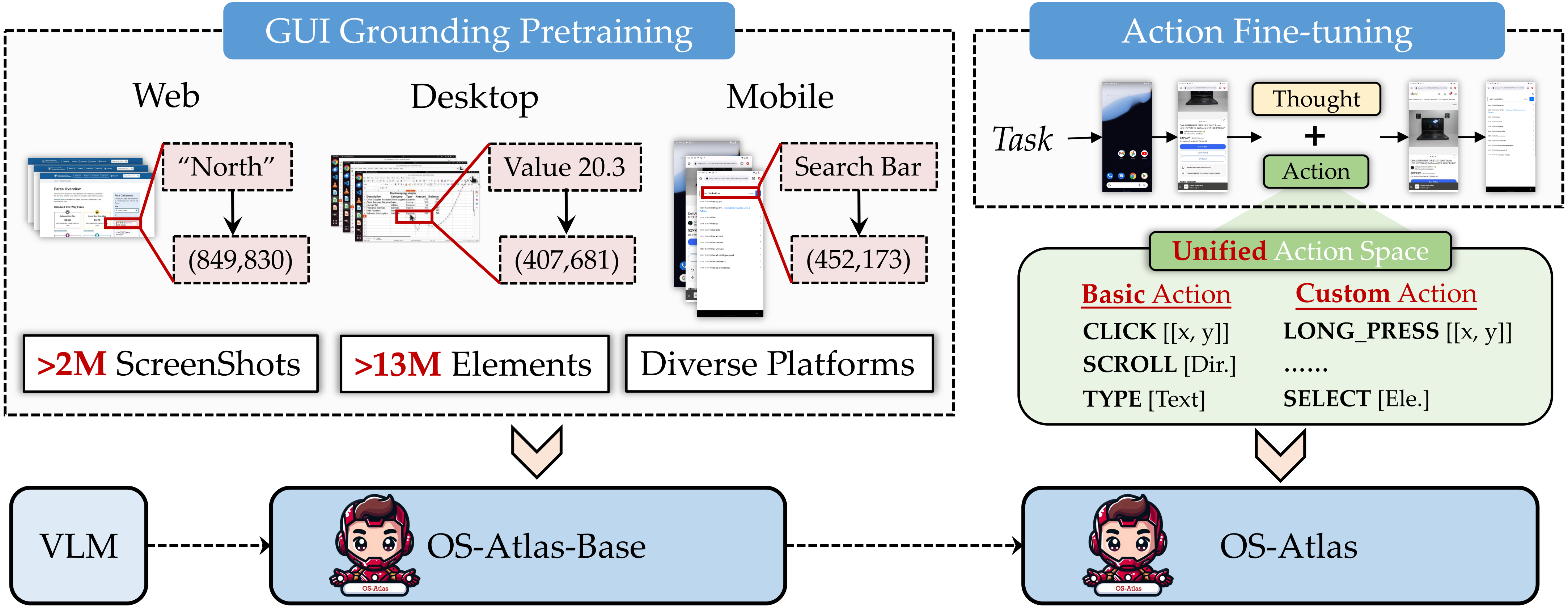
VLMs' Poor performance in GUI scenarios, because:

- Most existing VLMs are rarely pretrained on GUI screenshot images
- The heterogeneity of content and format in existing datasets

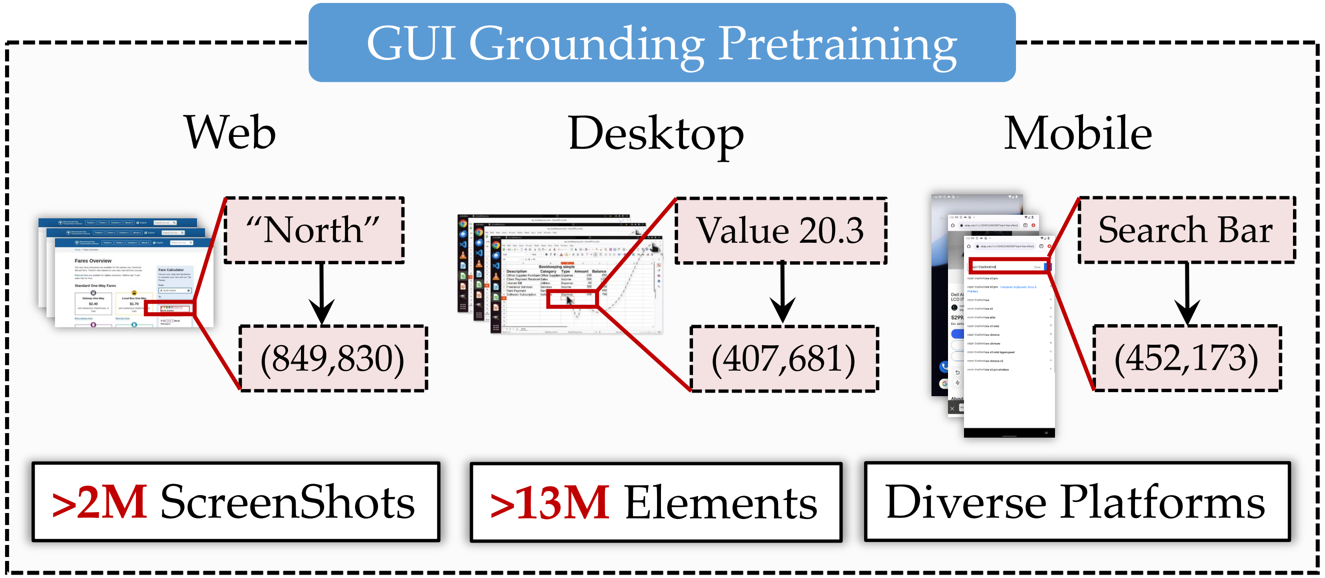


- Grounding Mode: Superior GUI Grounding and Plug-in with Planner
- Action Mode: Zero-shot Generalization on OOD tasks
- Agent Mode: DIY your own agent

Two-Stage Training



Infrastructure and Data Synthesis



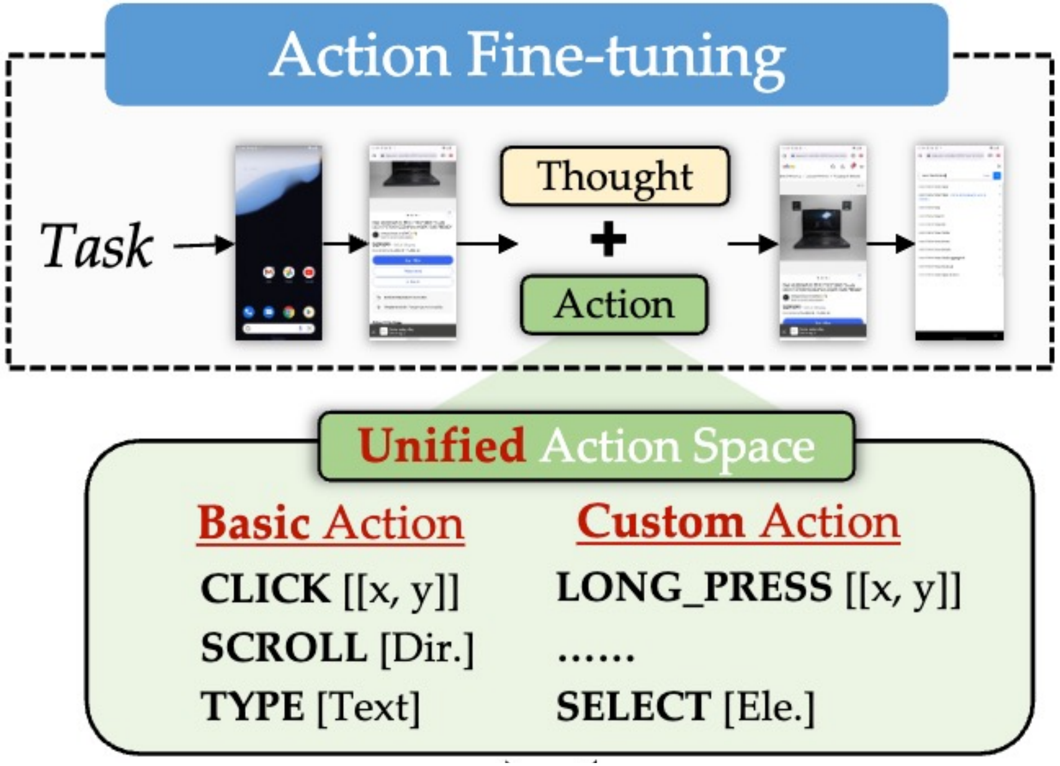
Dataset	#Screenshots Web	Mobile	Desktop	Open Source	#Elements
SeeClick	270K	94K	-	✓	3.3M
Ferret-UI	-	124K	-	✗	<1M
GUICourse	73K	9K	-	✓	10.7M
CogAgent	400K	-	-	✗	70M
OS-Atlas	1.9M	285K	54K	✓	13.58M

- The first multi-platform GUI grounding data synthesis toolkit, including：

- **Web -** Collected a large number of URLs from Common Crawl.
- **Desktop -** Windows, Linux and MacOS (integrated with OSWorld and uses random walk to collect trajectories).
- **Mobile -** Android (integrated with AndroidWorld).

- Training set comprises over 2.3 M distinct screenshots and more than 13 M GUI elements.

Action-Finetuning Stage



- OS-Atlas-Base → OS-Atlas
- Unified Action Space (Basic + Custom)
- Task-level Agent model

Experiments: GUI Grounding

Planner	Grounding Models	Mobile		Desktop		Web		Avg.
		Text	Icon/Widget	Text	Icon/Widget	Text	Icon/Widget	
-	Fuyu	41.00	1.30	33.00	3.60	33.90	4.40	19.50
	CogAgent	67.00	24.00	74.20	20.00	70.40	28.60	47.40
	SeeClick	78.00	52.00	72.20	30.00	55.70	32.50	53.40
	InternVL-2-4B	9.16	4.80	4.64	4.29	0.87	0.10	4.32
	Qwen2-VL-7B	61.34	39.29	52.01	44.98	33.04	21.84	42.89
	UGround-7B	82.80	60.30	82.50	63.60	80.40	70.40	73.30
	OS-Atlas-Base-4B	85.71	58.52	72.16	45.71	82.61	63.11	70.13
	OS-Atlas-Base-7B	**93.04**	**72.93**	**91.75**	**62.86**	**90.87**	**74.27**	**82.47**
GPT-4o	SeeClick	83.52	59.39	82.47	35.00	66.96	35.44	62.89
	UGround-7B	93.40	76.90	92.80	**67.90**	88.70	68.90	81.40
	OS-Atlas-Base-4B	**94.14**	73.80	77.84	47.14	86.52	65.53	76.81
	OS-Atlas-Base-7B	93.77	**79.91**	**90.21**	66.43	**92.61**	**79.13**	**85.14**

OS-Atlas-Base-7B achieves **SOTA** performance on ScreenSpot.

Experiments: Disentangled Planning and Action

Models	Successful Rate										Avg.
	OS	Calc	Impress	Writer	VLC	TB	Chrome	VSC	GIMP	WF	
GPT-4o + SoM	20.83	0.00	6.77	4.35	6.53	0.00	4.35	4.35	0.00	3.60	4.59
GPT-4o	8.33	0.00	6.77	4.35	16.10	0.00	4.35	4.35	3.85	5.58	5.03
+ SeeClick	16.67	0.00	12.76	4.35	23.52	6.67	10.86	8.70	11.54	7.92	9.21
+ OS-Atlas-Base-4B	20.83	2.23	14.89	8.70	23.52	13.33	15.22	13.04	15.38	7.92	11.65
+ OS-Atlas-Base-7B	25.00	4.26	17.02	8.70	29.41	26.67	19.57	17.39	19.23	8.91	14.63
Human	75.00	61.70	80.85	73.91	70.59	46.67	78.26	73.91	73.08	73.27	72.36

- GPT-4o: 5% on OSWorld
- GPT-4o + OS-Atlas: 14.6%

Insight: next bottleneck ? => complex reasoning and planning.

Experiments: Zero-shot and SFT

Web and Desktop

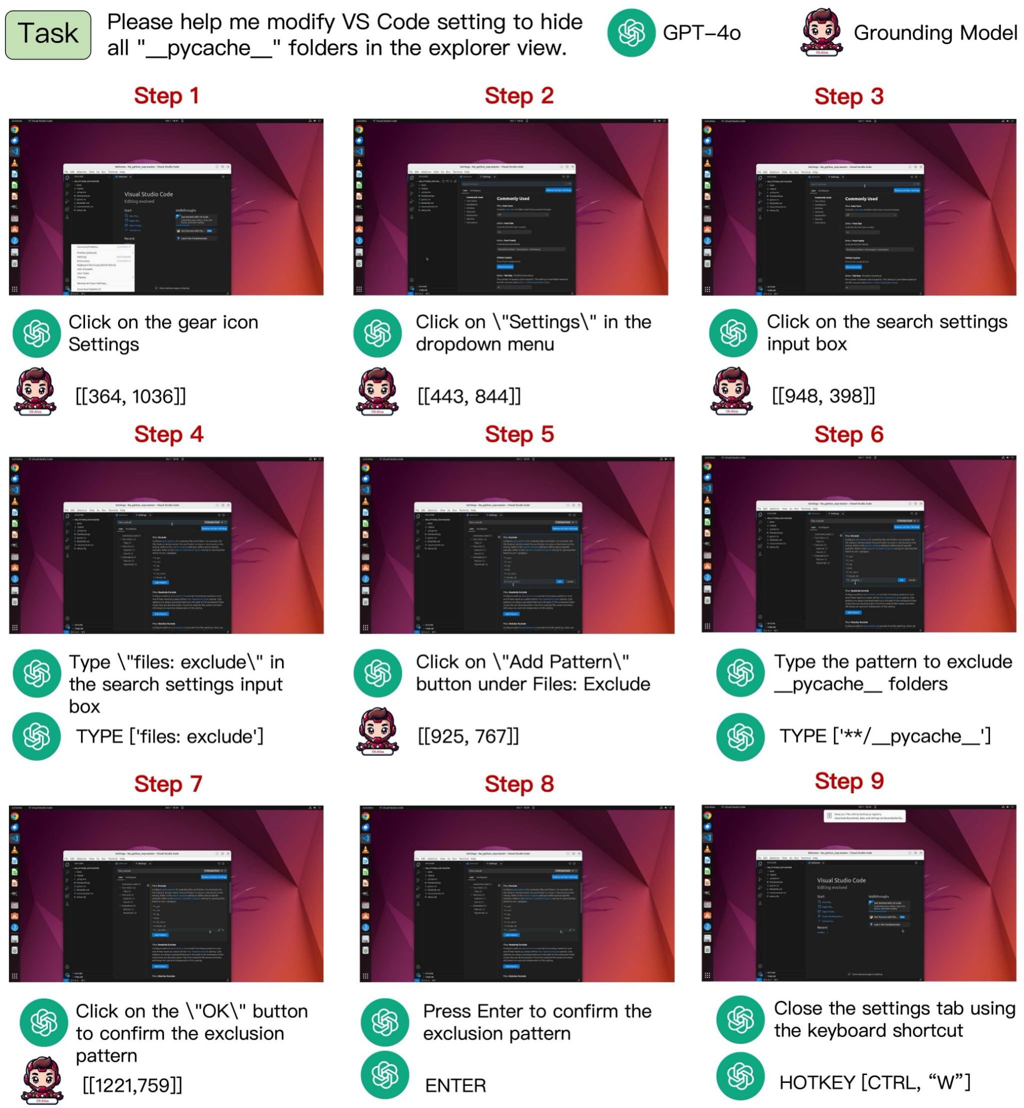
Models	GUI-Act-Web			OmniAct-Web			OmniAct-Desktop		
	Type	Grounding	SR	Type	Grounding	SR	Type	Grounding	SR
Zero-shot OOD Setting									
GPT-4o	77.09	45.02	41.84	79.33	42.79	34.06	79.97	63.25	50.67
OS-Atlas-4B	79.22	58.57	42.62	46.74	49.24	22.99	63.30	42.55	26.94
OS-Atlas-7B	**86.95**	**75.61**	**57.02**	**85.63**	**69.35**	**59.15**	**90.24**	**62.87**	**56.73**
Supervised Fine-tuning Setting									
InternVL-2-4B	81.42	47.03	36.17	47.51	51.34	24.39	67.00	44.47	29.80
Qwen2-VL-7B	89.36	90.66	82.27	89.22	85.94	78.58	96.27	94.52	91.77
SeeClick	88.79	78.59	72.34	86.98	75.48	68.59	96.79	70.22	72.69
OS-Atlas-4B	**89.36**	89.16	81.06	88.56	82.00	73.91	96.51	85.53	84.78
OS-Atlas-7B	89.08	**91.60**	**82.70**	**97.15**	**95.41**	**93.56**	**97.15**	**95.85**	**94.05**

Mobile

Models	AndroidControl-Low			AndroidControl-High			GUI-Odyssey		
	Type	Grounding	SR	Type	Grounding	SR	Type	Grounding	SR
Zero-shot OOD Setting									
GPT-4o	**74.33**	38.67	28.39	**63.06**	30.90	21.17	37.50	14.17	5.36
OS-Atlas-4B	64.58	71.19	40.62	49.01	49.51	22.77	49.63	34.63	20.25
OS-Atlas-7B	73.00	**73.37**	**50.94**	57.44	**54.90**	**29.83**	**60.42**	**39.74**	**26.96**
Supervised Fine-tuning Setting									
InternVL-2-4B	90.94	84.05	80.10	84.09	72.73	66.72	82.13	55.53	51.45
Qwen2-VL-7B	91.94	86.50	82.56	83.83	77.68	69.72	83.54	65.89	60.23
SeeClick	93.00	73.42	75.00	82.94	62.87	59.11	70.99	52.44	53.92
OS-Atlas-4B	91.92	83.76	80.64	84.69	73.79	67.54	83.47	61.37	56.39
OS-Atlas-7B	**93.61**	**87.97**	**85.22**	**85.22**	**78.48**	**71.17**	**84.47**	**67.80**	**61.98**

- OS-Atlas achieved SOTA performance across 3 different platforms, 6 distinct datasets, and 2 evaluation settings (Zero-shot OOD and SFT).
- Huge superiority over GPT-4o under zero-shot OOD setting.

More Cases about GUI Tasks



- OS-World
- Planner: GPT-4o
- Action Model: OS-Atlas (7B)

More Demos ?

Please check out our homepage !

- https://osatlas.github.io/

中文解读（OS-ATLAS）

Next

We already have strong action / foundation models that map instructions to actions.

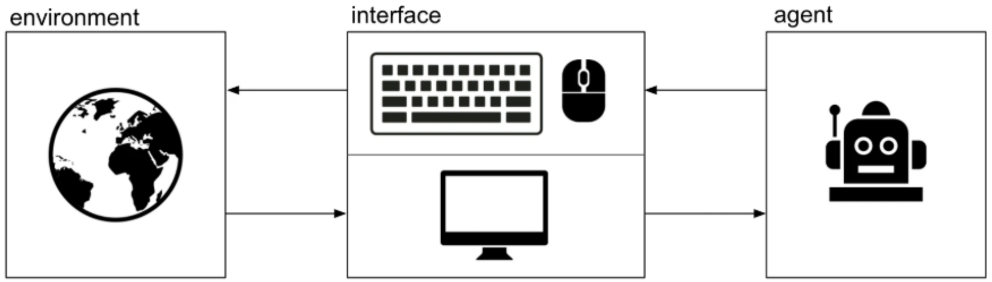
Now, we aim to empower agents with complete **Perception–Decision–Execution** capabilities.

Build Computer-using Agents

Quite promising to achieve digital automation in one model.

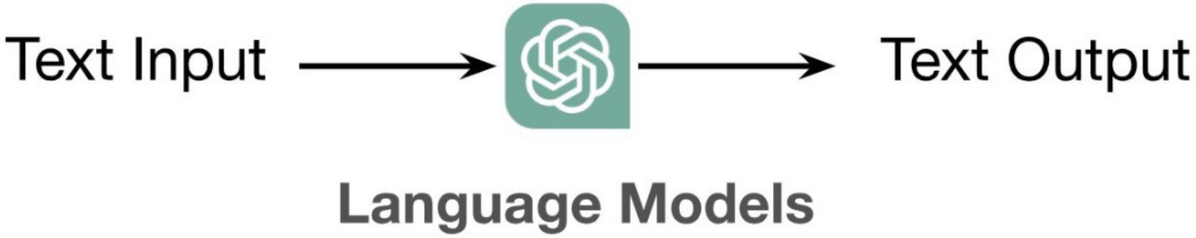
Can we transform a (V)LM into such GUI agents?

1. Perceive
2. Planning
3. Action

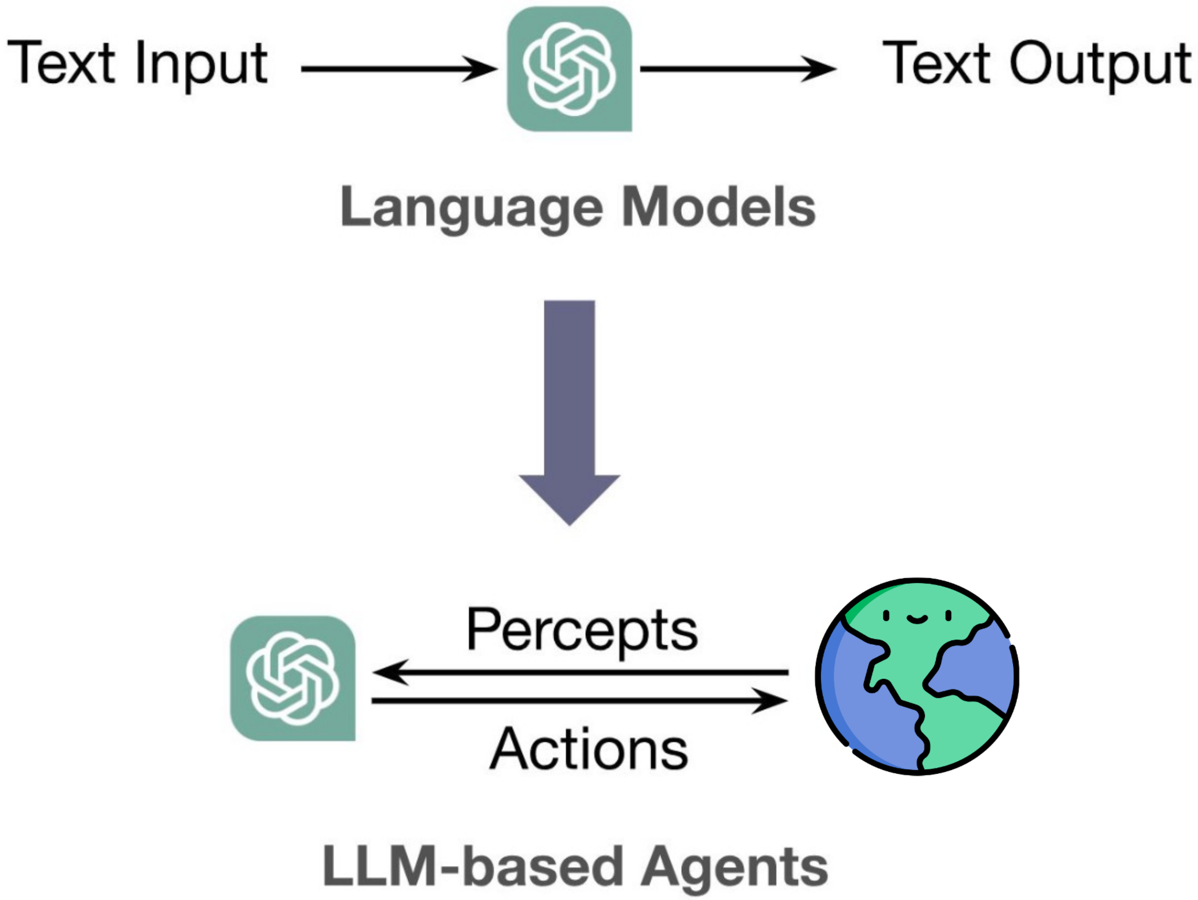


Of course! But it is a non-trivial job!

Recap: Language Agents



Recap: Language Agents



But this is not enough for Computer-using / GUI Agents.

Computer-using Agents

Agents are promising, but building powerful agents is challenging.

1. Agents need to follow human instructions.
2. Agents need to perform planning and action.
3. Agents need to perceive envs. and the applications they are interacting with.

Best Way to build Computer-using Agents

Behavioral Cloning / Imitation Learning.

Sounds good, but where is our data?

Data Problems

Human annotation for GUI data is much more expensive than you think.

Not to mention scenario/domain - specific data.

How about having the machine collect data?

1. Pre-defined tasks are required, but they may not align with the environment.
2. Limited diversity and a poor success rate.

Data Scarcity

So, our goals are as follows:

1. Eliminate human involvement.
2. Obtain high-quality Trajectory data.
3. Diversity and Scalability.

OS-Genesis Automating GUI Agent Trajectory Construction via Reverse Task Synthesis

ACL 2025 VIENNA

Qiushi Sun*, Kanzhi Cheng*, Zichen Ding*, Chuanyang Jin*, Yian Wang
Fangzhi Xu, Zhenyu Wu, Liheng Chen, Chengyou Jia, Zhoumianze Liu
Ben Kao, Guohao Li, Junxian He, Yu Qiao, Zhiyong Wu

上海人工智能实验室 Shanghai Artificial Intelligence Laboratory
Johns Hopkins University
Shanghai Jiao Tong University
University of Oxford

GUI Trajectory Data

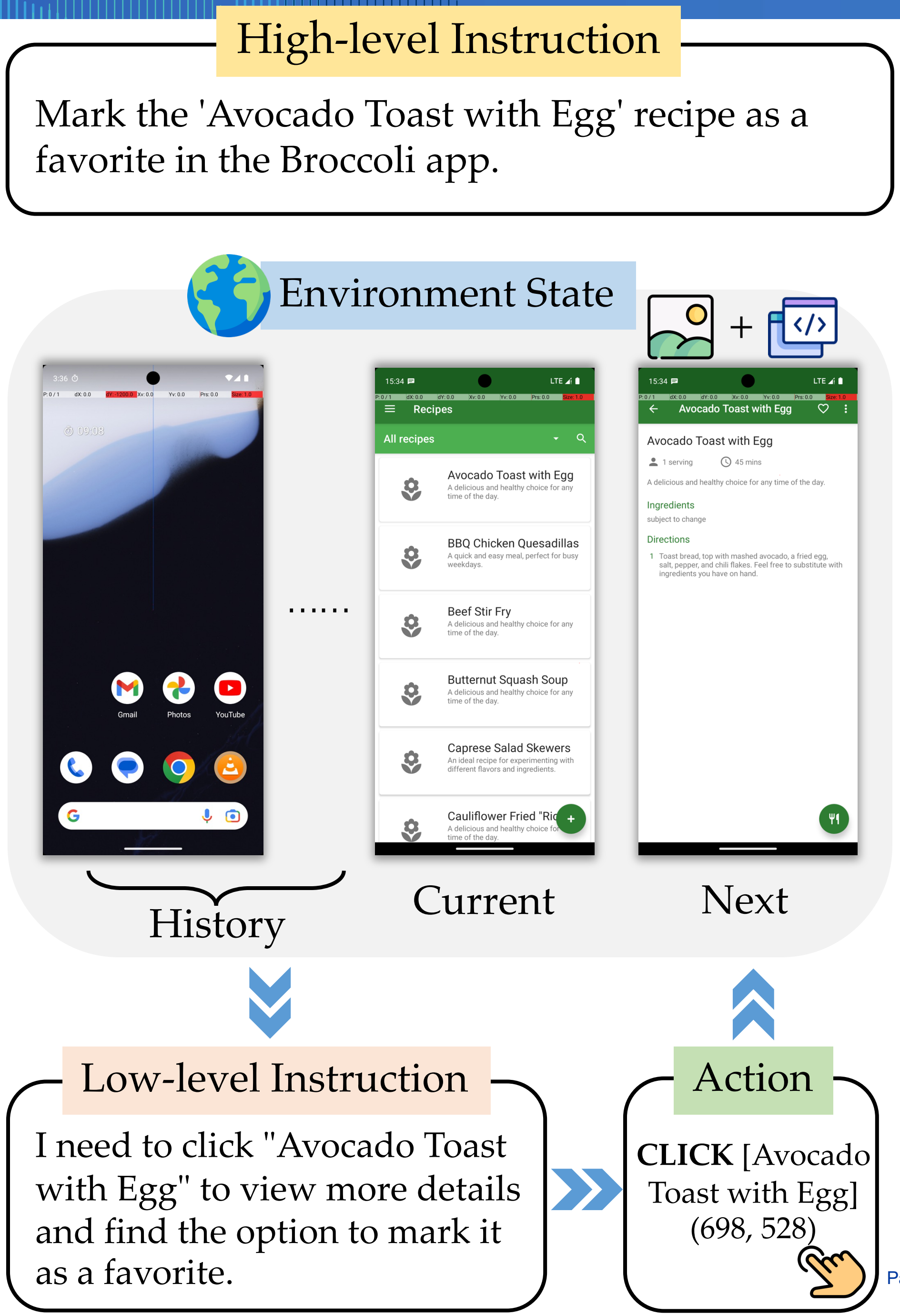
The best data format for GUI agents

1. A high-level instruction that defines the overall goal the agent aims to accomplish

2. A series of low-level instructions that each describe specific steps required

3. Actions (e.g., `CLICK`, `TYPE`)

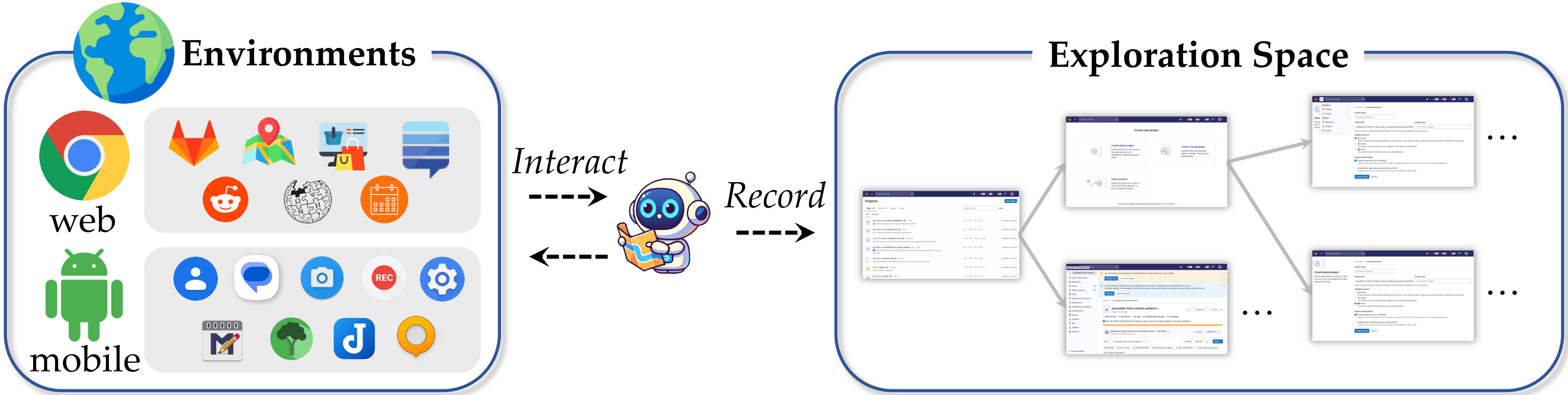
4. States, which include visual representations like screenshots and textual representations such as a11ytree



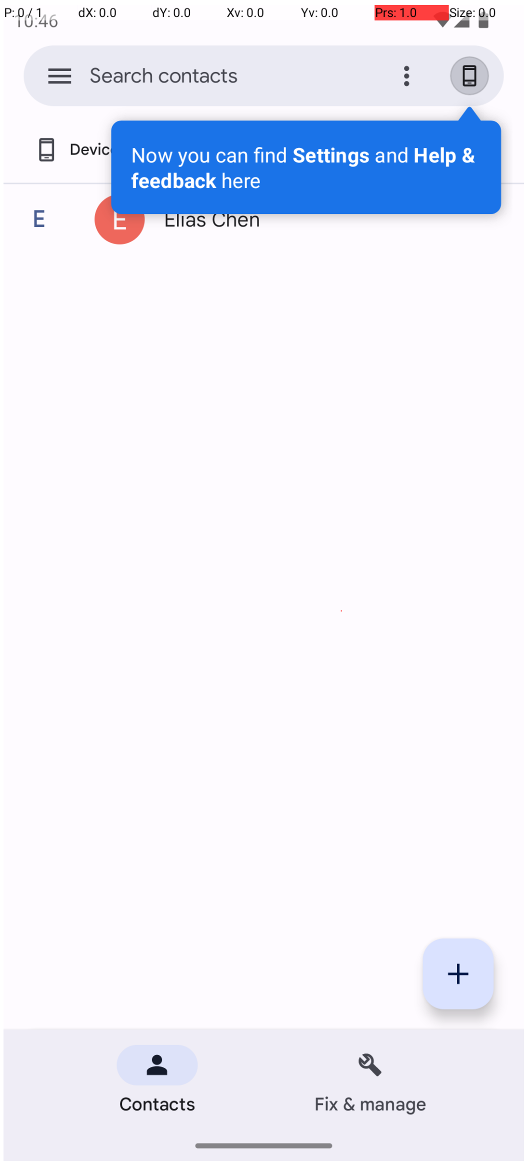
Reverse Task Synthesis

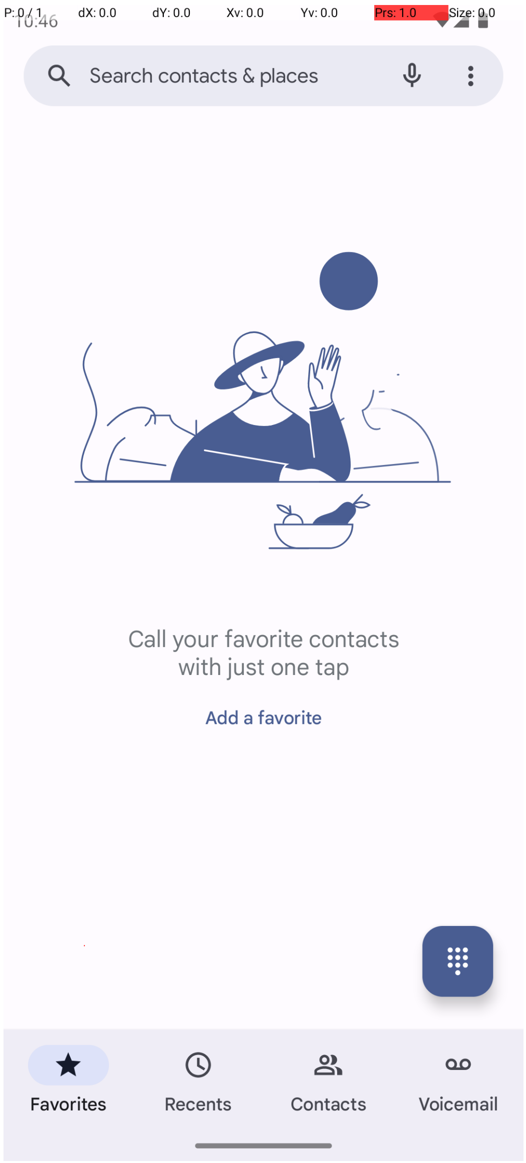
Interaction-Driven Functional Discovery is a rule-based process that explores dynamic GUI environments by interacting with UI elements. It uncovers functionalities through interaction triples

We collect: <Screen1, action, Screen2>



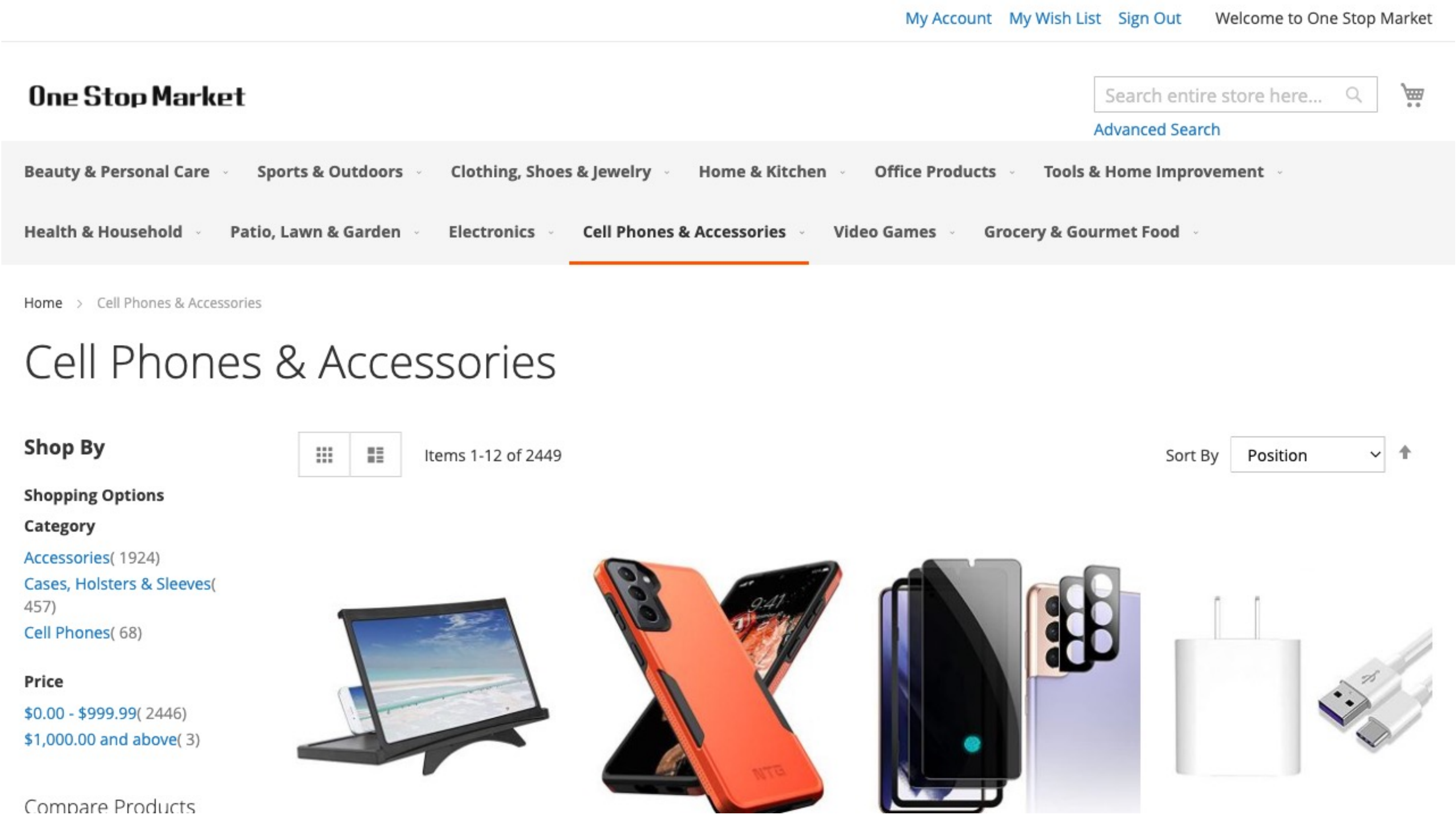
Dynamic Environments





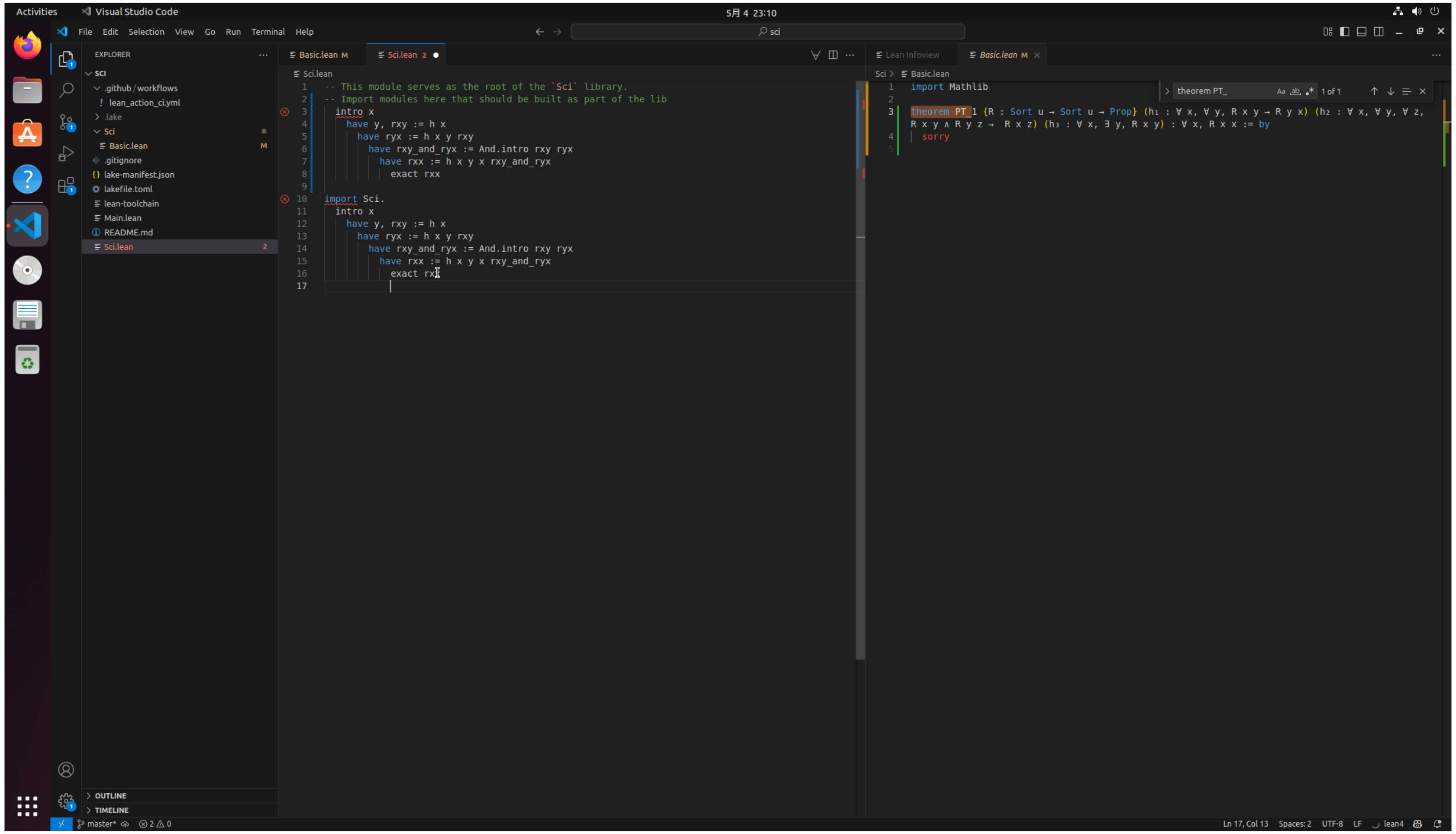
[11] AndroidWorld: A Dynamic Benchmarking Environment for Autonomous Agents, ICLR 2025

Dynamic Environments



[12] *WebArena: A Realistic Web Environment for Building Autonomous Agents, ICLR 2024*

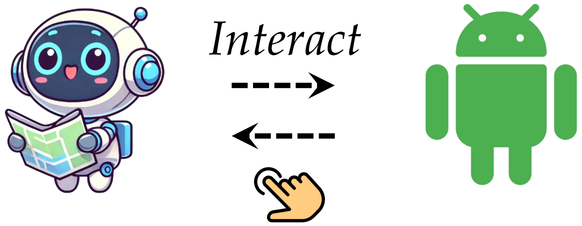
Dynamic Environments



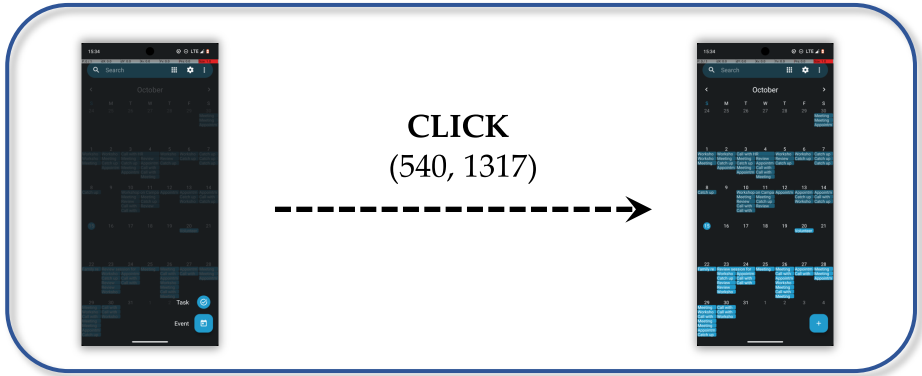
[6] ScienceBoard: Evaluating Multimodal Autonomous Agents in Realistic Scientific Workflows

Reverse Task Synthesis

Retroactively interpreting changes in the GUI environment caused by actions.

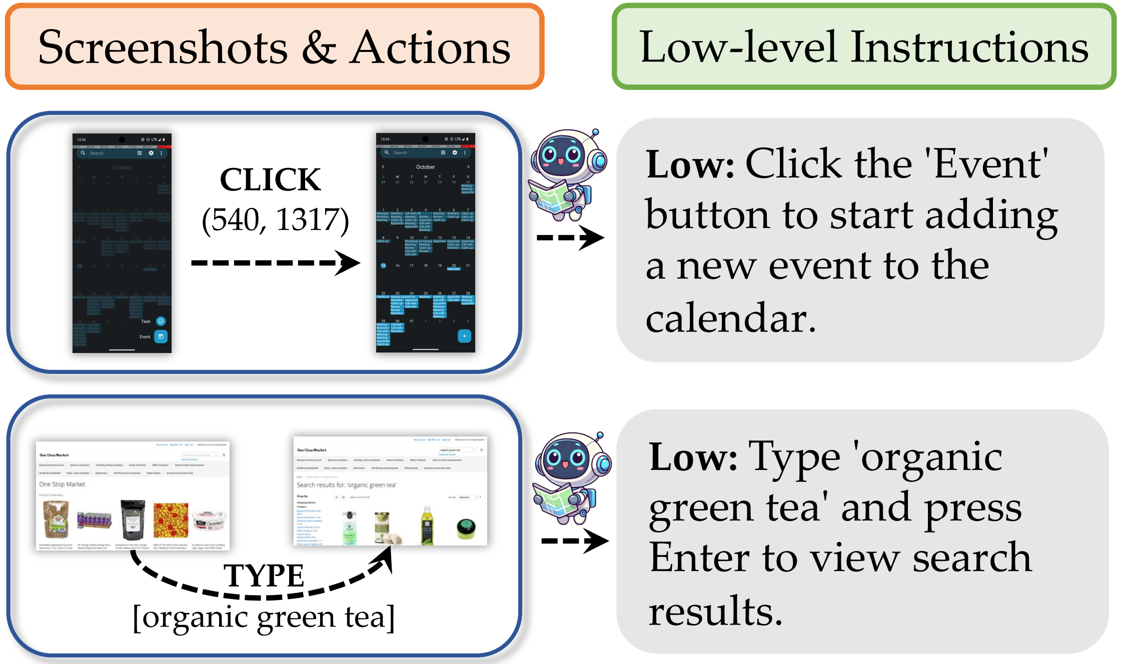


Screenshots & Actions



Reverse Task Synthesis

Retroactively interpreting changes in the GUI environment caused by actions, this process generates executable low-level instructions



The data we synthesized:

1. Grounded

2. Actionable

Reverse Task Synthesis

Retroactively interpreting changes in the GUI environment caused by actions, this process generates executable low-level instructions, which are then transformed into broader, goal-oriented high-level tasks

Screenshots & Actions	Low-level Instructions	High-level Instruction
	**Low:** Click the 'Event' button to start adding a new event to the calendar.	**High**: In Simple Calendar Pro, create a new event titled 'Team Meeting' scheduled for October 15, 2023, at 10:00 AM. Save the event after filling in the details.
	**Low:** Type 'organic green tea' and press Enter to view search results.	**High**: Search for 'organic green tea' and filter the results to show only products under the 'Health & Household' category, sorted by price from lowest to highest.

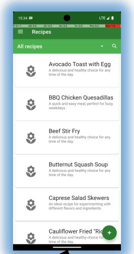
Reverse Task Synthesis

After reverse task synthesis generates task instructions, they are automatically executed in the GUI environment to build complete trajectories.

High: Mark the 'Avocado Toast with Egg' recipe as a favorite in the Broccoli app.

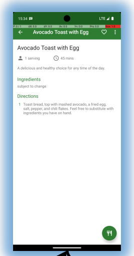
Low: I need to first open the Broccoli-Recipe App.

Action: OpenApp ("Broccoli")



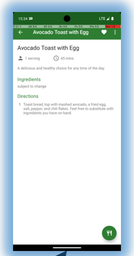
Low: Click on Avocado Toast with Egg to view more details.

Action: CLICK (698,528)



Low: Click "Mark as favorite"option.

Action: CLICK (912,201)

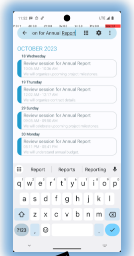


High: Set a reminder for the 'Review session for Annual Report' scheduled on October 18th in Simple Calendar Pro and save the changes.

Low: I need to first open the Simple Calendar Pro.

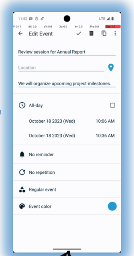
Action: OpenApp ("Simple Calendar Pro")

……



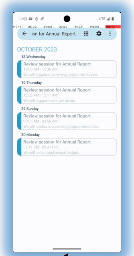
Low: Select the specific event "Review session for Annual Report".

Action: CLICK (77, 596)



Low: Click "Save" button.

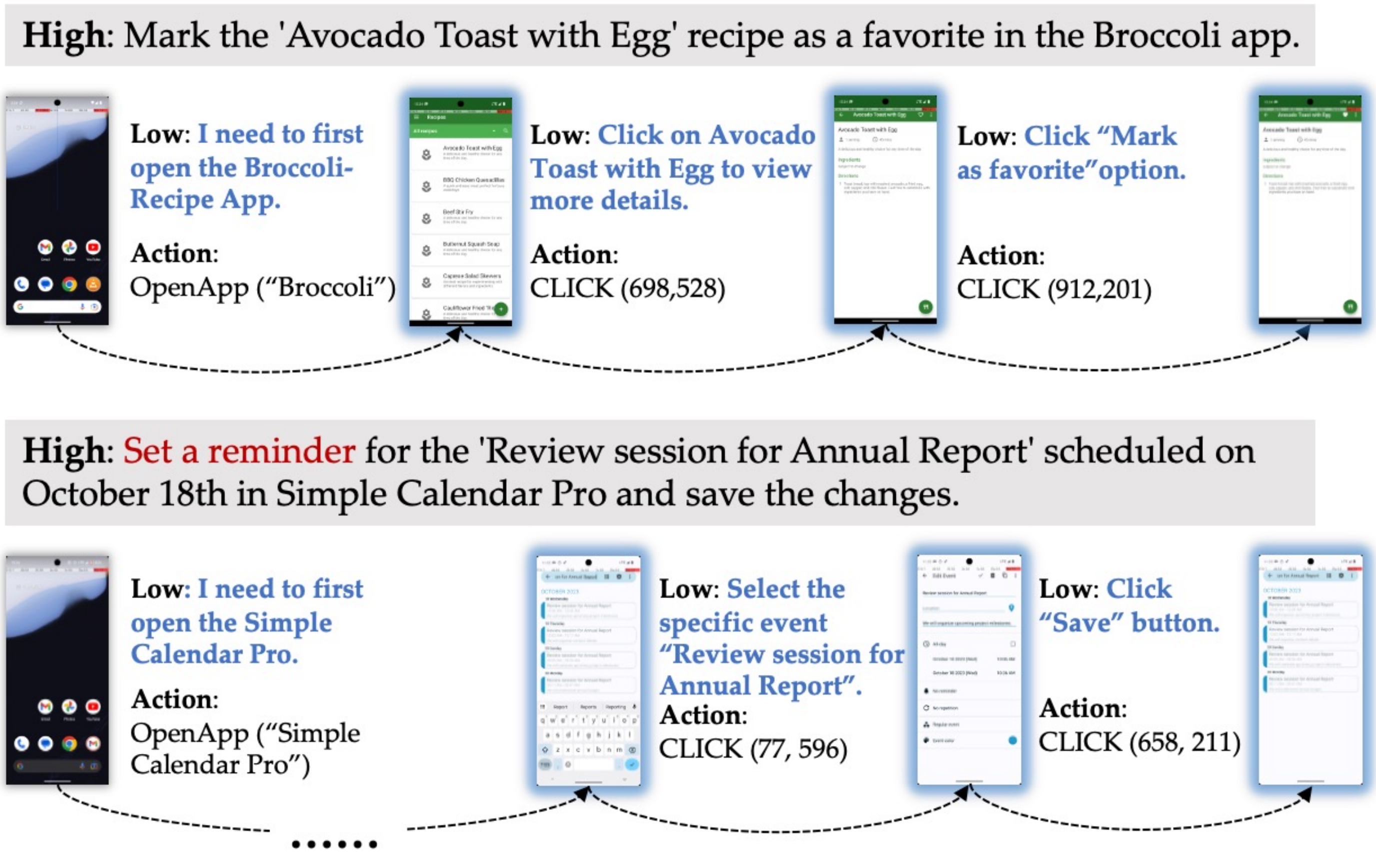
Action: CLICK (658, 211)

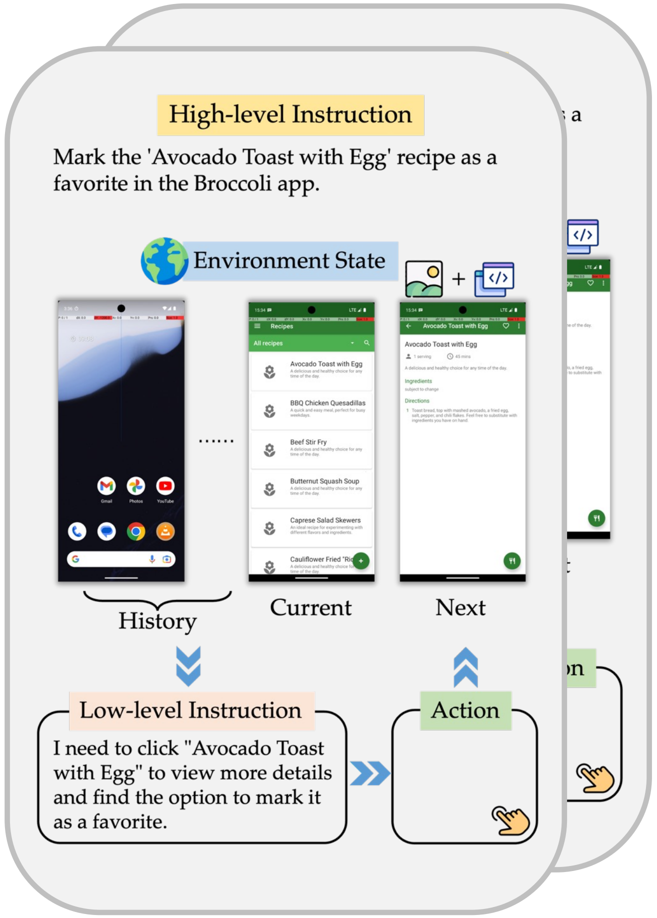


Reverse Task Synthesis

Trajectories collected! But is this all?

Let's consider data quality and synthesis efficiency.





Data Quality Control

Tasks are executed by machines, not all of them are successful.

Previous approach:

1. Training all data at once - what about the quality?

2. Discarding all incomplete Trajectories - what about the efficiency?

Thus, we introduce a Trajectory Reward Model to handle this.

Reward Modeling

We introduce a Trajectory Reward Model for weighted sampling in training.

Models

Data Synthesis

GPT-4o

Qwen-VL Qwen2-VL-72B-Instruct

Backbones

InternVL InternVL2-4B / 8B

Qwen-VL Qwen2-VL-7B-Instruct

[13] *Qwen2.5-VL Technical Report*

[14] *Expanding Performance Boundaries of Open-Source Multimodal Models with Model, Data, and Test-Time Scaling*

Training Strategies

Leverage trajectory characteristics to train GUI agents with complete capabilities

1. Planning Training

High-level   Low-level

$$\mathcal{L}_1 = -\sum_{t_i \in \mathcal{T}} \log\left(p_\theta(\ell \mid s, h_i, c) \cdot p_\theta(a \mid s, h_i, c, \ell)\right)$$

2. Action Training

Input   Actions   History

$$\mathcal{L}_2 = -\sum_{t_i \in \mathcal{T}} \log p_\theta(a \mid s, c, \ell)$$

Training Strategies

After Training, our agents will generate ReACT-Style output

Examples:

Step 1: To create a new folder in Markor, I need to first open the Markor app.

action: `{"action_type":"open_app", "app_name":"Markor"}`

Step 2: To create a new folder, I need to click on the "Create a new file or folder" button, which is indicated by the plus icon.

action: `{"action_type":"click","x":964.5,"y":2074.5}`

Step 3: I need to change the folder name to folder_20241224. The current text field for the folder name is visible and editable.

action:
`{"action_type":"type","text":"folder_20241224","x":373.5,"y":552.0}`

[15] ReAct: Synergizing Reasoning and Acting in Language Models, ICLR 2023

Baselines

We adapt / build the following forward baselines

- **Zero-Shot.** Advanced prompting-based agents, such as M3A.

- **Task-Driven.** GUI Trajectories synthesized using pre-defined tasks. Given initial screenshots of the app/web page and task examples, use GPT-4 to generate high-level instructions and collect data.

- **Self-Instruct.** Builds on Task-Driven by adding self-instructed tasks.

Setting: Screenshot + A11ytree

[16] Self-Instruct: Aligning Language Models with Self-Generated Instructions, ACL 2023

Experiments: Mobile

Base Model	Strategies	AndroidWorld	AndroidControl-High		AndroidControl-Low	
			SR	Type	SR	Type
GPT-4o	Zero-Shot (M3A)	23.70	53.04	69.14	69.59	80.27
InternVL2-4B	Zero-Shot	0.00	16.62	39.96	33.69	60.65
	Task-Driven	4.02	27.37	47.08	66.48	90.37
	Task-Driven w. Self Instruct	7.14	24.95	44.27	66.70	90.79
	OS-Genesis	**15.18**	**33.39**	**56.20**	**73.38**	**91.32**
InternVL2-8B	Zero-Shot	2.23	17.89	38.22	47.69	66.67
	Task-Driven	4.46	23.79	43.94	64.43	89.83
	Task-Driven w. Self Instruct	5.36	23.43	44.43	64.69	89.85
	OS-Genesis	**16.96**	**35.77**	**64.57**	**71.37**	**91.27**
Qwen2-VL-7B	Zero-Shot	0.89	28.92	61.39	46.37	72.78
	Task-Driven	6.25	38.84	58.08	71.33	88.71
	Task-Driven w. Self Instruct	9.82	39.36	58.28	71.57	89.73
	OS-Genesis	**17.41**	**44.54**	**66.15**	**74.17**	**90.72**

Table 1: Performance on AndroidWorld and AndroidControl benchmarks.

Findings: OS-Genesis + Opensource VLM > Propriety Models + Complex Prompting

Experiments: Web

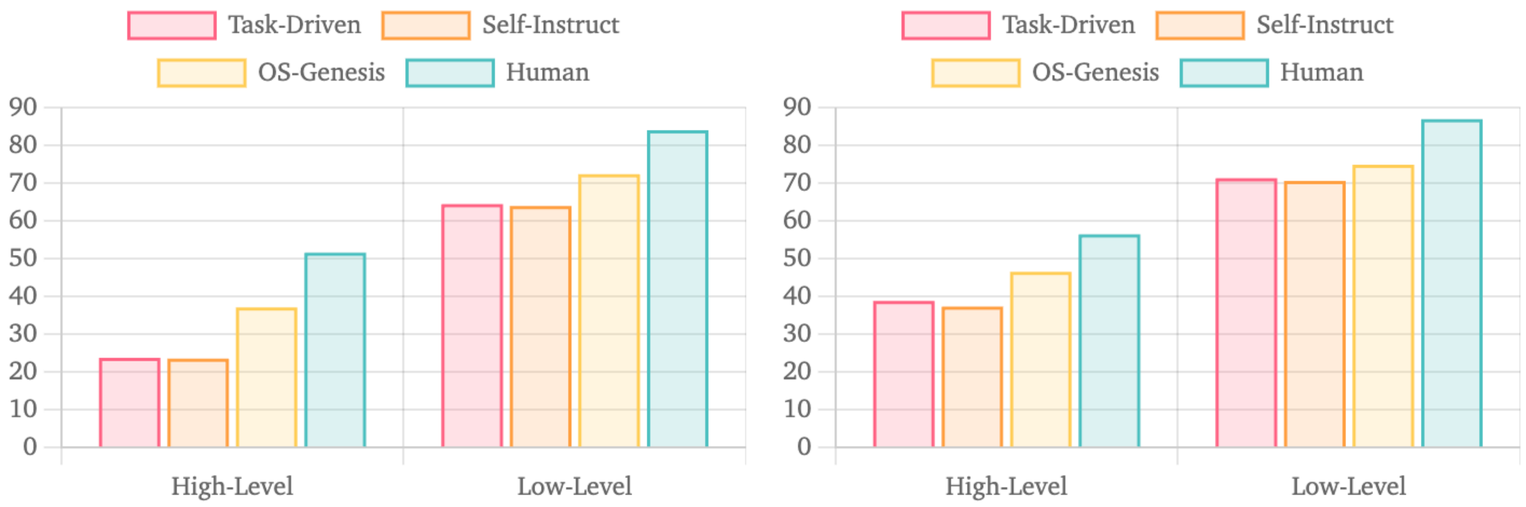
Base Model	Strategies	Shopping	CMS	Reddit	Gitlab	Maps	Overall
GPT-4o	Zero-Shot	14.28	21.05	6.25	14.29	20.00	16.25
InternVL2-4B	Zero-Shot	0.00	0.00	0.00	0.00	0.00	0.00
	Task-Driven	5.36	1.76	0.00	**9.52**	5.00	4.98
	Task-Driven w. Self Instruct	5.36	3.51	0.00	**9.52**	7.50	5.81
	OS-Genesis	**10.71**	**7.02**	**3.13**	7.94	**7.50**	**7.88**
InternVL2-8B	Zero-Shot	0.00	0.00	0.00	0.00	0.00	0.00
	Task-Driven	3.57	7.02	0.00	6.35	2.50	4.56
	Task-Driven w. Self Instruct	**8.93**	10.53	6.25	**7.94**	0.00	7.05
	OS-Genesis	7.14	**15.79**	**9.34**	6.35	**10.00**	**9.96**
Qwen2-VL-7B	Zero-Shot	**12.50**	7.02	6.25	6.35	5.00	7.47
	Task-Driven	8.93	7.02	6.25	6.35	5.00	7.05
	Task-Driven w. Self Instruct	8.93	1.76	3.13	4.84	**7.50**	5.39
	OS-Genesis	7.14	**8.77**	**15.63**	**15.87**	5.00	**10.79**

Table 2: Performance on WebArena benchmarks.

Analysis

How Far are we from Human Data?

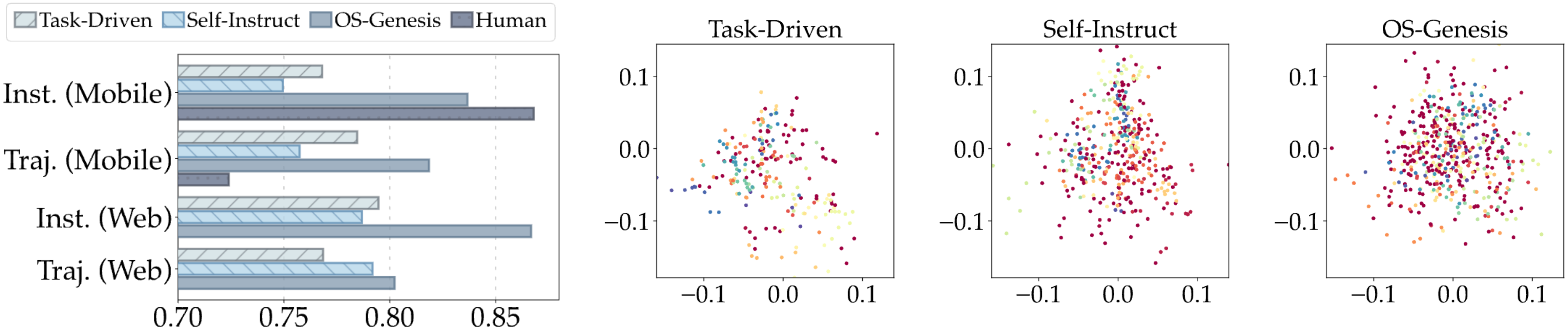
Then, OS-Genesis v.s. Human-annotated Trajectories.



Insight: OS-Genesis achieves ~80% of human data's effectiveness.

Analysis

How about our data diversity?



Insight: Significantly better than Forward methods and approaches the human level.

Checkpoints & Data Access

Available on HuggingFace

OS-Genesis: Automating GUI Agent Trajectory Construction via Reverse Task Synthesis

Published on Dec 28, 2024 · Submitted by QiushiSun on Jan 2 · #1 Paper of the day

Authors: Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, Zhiyong Wu

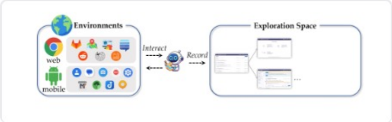
Abstract

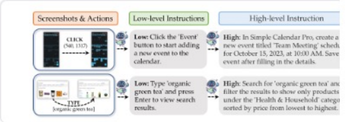
OS-Genesis is a novel GUI data synthesis pipeline that enhances the training of GUI agents by reversing the trajectory collection process to improve data quality and diversity.

AI-generated summary

Graphical User Interface (GUI) agents powered by Vision-Language Models (VLMs) have demonstrated human-like computer control capability. Despite their utility in advancing digital automation, a critical bottleneck persists: collecting high-quality trajectory data for training. Common practices for collecting such data rely on human supervision or synthetic data generation through executing pre-defined tasks, which are either resource-intensive or unable to guarantee data quality. Moreover, these methods suffer from limited data diversity and significant gaps between synthetic data and real-world environments. To address these challenges, we propose OS-Genesis, a novel GUI data synthesis pipeline that reverses the conventional trajectory collection process. Instead of relying on pre-defined tasks, OS-Genesis enables agents first to perceive environments and perform step-wise interactions, then retrospectively derive high-quality tasks to enable trajectory-level exploration. A trajectory reward model is then employed to ensure the quality of the generated trajectories. We demonstrate that training GUI agents with OS-Genesis significantly improves their performance on highly challenging online benchmarks. In-depth analysis further validates OS-Genesis's efficiency and its superior data quality and diversity compared to existing synthesis methods. Our codes, data, and checkpoints are available at https://qiushisun.github.io/OS-Genesis-Home/{OS-Genesis Homepage}.

View arXiv page · View PDF · Project page · GitHub ★ 146 · Add to collection





Community

QiushiSun · Paper author · Paper submitter · Jan 2

This paper introduces OS-Genesis, an interaction-driven pipeline for synthesizing high-quality and diverse GUI agent trajectory data without human supervision or predefined tasks. By leveraging reverse task synthesis and a trajectory reward model, OS-Genesis enables effective end2end training of GUI agents.

👍 5 · Reply

Upvoted 88 · +76

Models citing this paper 9

- OS-Copilot/OS-Genesis-7B-AC — Image-Text-to-Text · 8B · Updated Jan 8 · 58 · 7
- OS-Copilot/OS-Genesis-4B-AC — Image-Text-to-Text · 4B · Updated Jan 8 · 31 · 7
- OS-Copilot/OS-Genesis-8B-AC — Image-Text-to-Text · 8B · Updated Jan 8 · 38 · 4
- OS-Copilot/OS-Genesis-7B-AW — Any-to-Any · 8B · Updated May 5 · 28 · 1

Browse 9 models citing this paper

Datasets citing this paper 2

- OS-Copilot/OS-Genesis-mobile-data — Viewer · Updated Mar 17 · 51.1k · 201 · 2
- OS-Copilot/OS-Genesis-web-data — Updated Mar 17 · 54 · 3

Spaces citing this paper 0

No Space linking this paper

Cite arxiv.org/abs/2412.19723 in a Space README.md to link it from this page.

Collections including this paper 17

- UI Agent · Collection — a collection of algorithmic agents for user... · 382 items · Updated about 3 hours ago · 57
- Papers · Collection — 540 items · Updated 3 days ago · 11
- Synthetic Data and Self-Improvement · Collection — 82 items · Updated Apr 24 · 7

Our Project

OS-Genesis

Automating GUI Agent Trajectory Construction via Reverse Task Synthesis

Introducing OS-Genesis, a ***manual-free*** data pipeline for synthesizing GUI agent trajectory. OS-Genesis is characterized by the following core features:

- **Interaction-driven**: Agents actively explore GUI environments through stepwise interactions to discover functionalities and generate data.
- **Reverse Task Synthesis**: OS-Genesis retroactively derives meaningful low/high-level task instructions from observed interactions and state changes, enabling the construction of diverse and executable trajectories without pre-defined tasks.
- **Trajectory Data**: We construct and release high-quality mobile and web trajectories to accelerate GUI agents research.
- **Performance**: OS-Genesis significantly outperforms other synthesis methods on benchmarks like AndroidWorld and WebArena.

arXiv | Code | Checkpoints | Data

中文解读（OS-Genesis）

https://qiushisun.github.io/OS-Genesis-Home/

Another Solution for Data Scarcity?

OS-Genesis is cool! 

However, there are still limitations — for example, the type of synthetic data is constrained by the environment itself.

A single environment may reach its limit after producing just tens of 10K samples.

Can we push it even further?

GUI Trajectory Data

Issue: Although we have collected more trajectory data, it still remains limited compared to general LLM/VLM tasks.

Domains	Datasets	Samples	Type
Web	OS-Genesis (Web) (Sun et al., 2024b)	3,789	Instruction, Thought, Action
	MM-Mind2Web (Zheng et al., 2024a)	21,542	Instruction, Thought, Action
	VisualWebArena (Koh et al., 2024a)	3,264	Instruction, Thought, Action
Mobile	OS-Genesis (Mobile) (Sun et al., 2024b)	4,941	Instruction, Thought, Action
	Aguvis (Xu et al., 2024b)	22,526	Instruction, Thought, Action

Table 2: Statistics of the web/mobile domains along with the corresponding GUI trajectory datasets used in post-training.

RQ: Is it possible to leverage "external forces" to further enhance the use of GUI data?

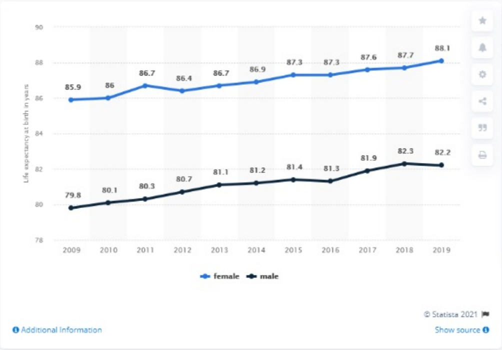
Breaking the Data Barrier – Building GUI Agents Through Task Generalization

Junlei Zhang*; Zichen Ding*, Chang Ma, Zijie Chen, Qiushi Sun,
Zhenzhong Lan, Junxian He

Enhancing GUI Agent with Non-GUI Data

However, we have abundant non-GUI data available to enhance versatile abilities, such as complex reasoning

Can we take advantage of these data-rich domains?

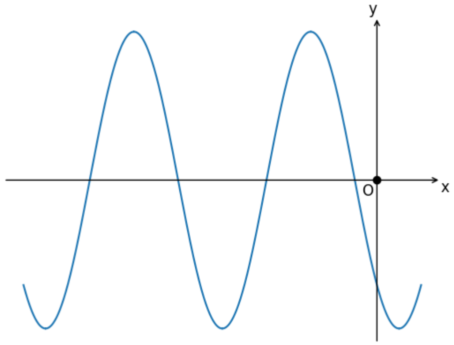


Chart

Prove that the sum of the squares of the lengths of the medians of a tetrahedron is equal to 4/9 of the sum of the squares of the lengths of its edges.

Text Math

Embodied

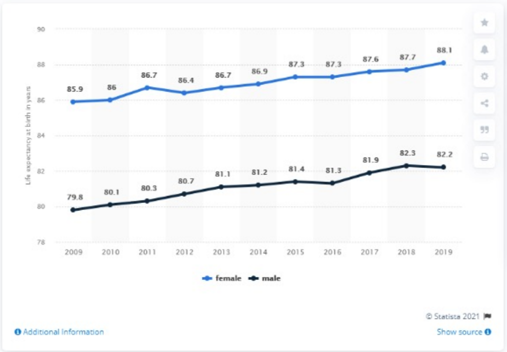


Multi-modal Math

Enhancing GUI Agent with Non-GUI Data

We introduce **Mid-Training** to the GUI Agent training:

Mid-Training refers to the training phrase between pre-training and post-training, enhance the fundamental abilities of models

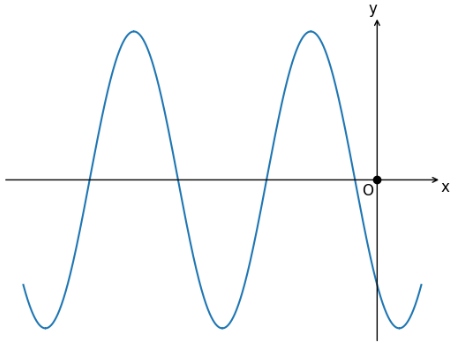


Chart

Prove that the sum of the squares of the lengths of the medians of a tetrahedron is equal to 4/9 of the sum of the squares of the lengths of its edges.

Text Math

Embodied



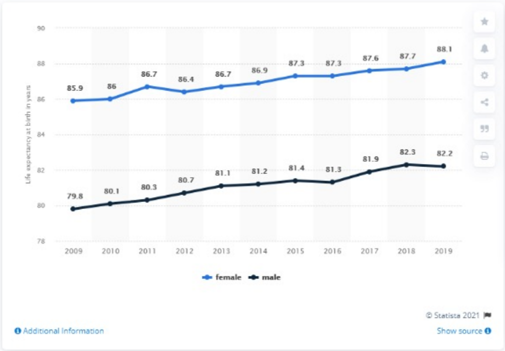
Multi-modal Math

Enhancing GUI Agent with Non-GUI Data

Mid-training with Non-GUI data:

1. Naively training on non-GUI data, then post-training on GUI data can lead to gradient conflicts.

2. What kinds of domains should we use?

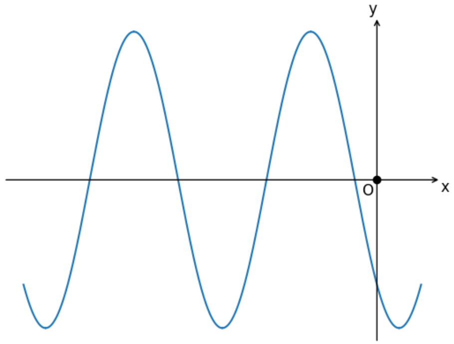


Chart

Prove that the sum of the squares of the lengths of the medians of a tetrahedron is equal to 4/9 of the sum of the squares of the lengths of its edges.

Text Math

Embodied



Multi-modal Math

Enhancing GUI Agent with Non-GUI Data

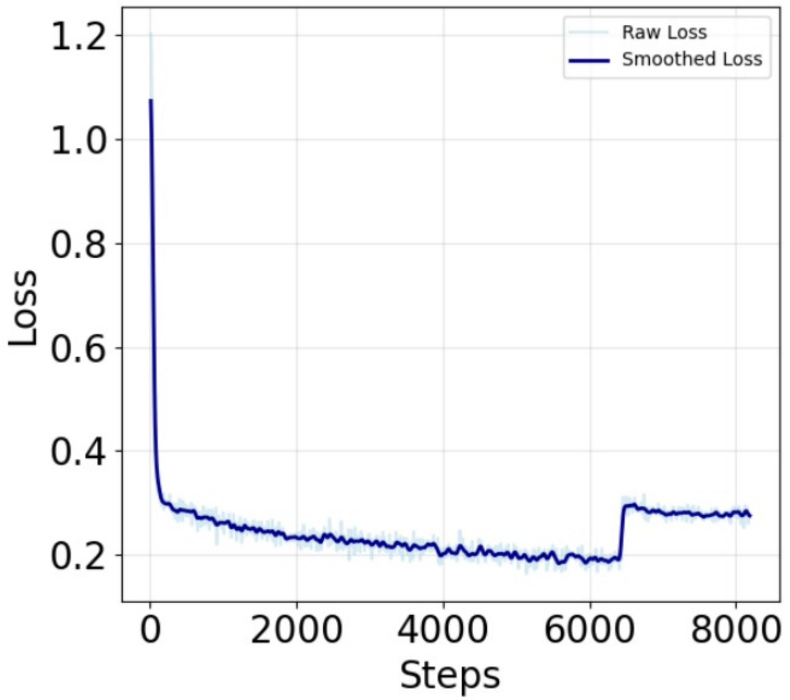
So, our goals are as follows:

1. Discover generalizable non-GUI domains
2. Design stable training methods.
3. Combine the generalizable to obtain larger mid-training dataset.

Mid-Training

1. We concatenate mid-training data with GUI trajectory and train sequentially. Both stages are integrated under a single optimizer and learning rate.

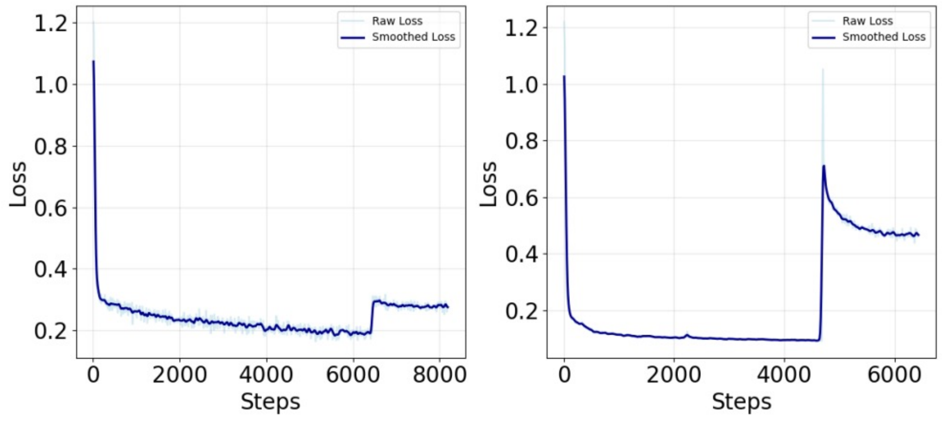
2. We mix the GUI trajectory into the mid-training data during the mid-training stage, to stabilize the training.



Mid-Training

1. We concatenate mid-training data with GUI trajectory and train sequentially. Both stages are integrated under a single optimizer and learning rate.

2. We mix the GUI trajectory into the mid-training data during the mid-training stage, to stabilize the training.



(a) Multi-modal Math w/ mixing

(b) Multi-modal Math w/o mixing

Mid-Training

We adapt the following baselines:

- **Fine-tuned Qwen2-VL-7B-Instruct.** We post-train Qwen2-VL-7B-Instruct directly as the baseline.

- **GPT-4o.**

Mid-Training

Domains	Observation	WebArena PR	WebArena SR	AndroidWorld SR
GUI Post-Training Only	Image	26.3	6.2	9.0
Public Baselines				
GPT-4o-2024-11-20	Image	36.9	15.6	11.7
OS-Genesis-7B	Image + Accessibility Tree	–	–	17.4
AGUVIS-72B	Image	-	-	26.1
Claude3-Haiku	Accessibility Tree	26.8	12.7	-
Llama3-70b	Accessibility Tree	35.6	12.6	-
Gemini1.5-Flash	Accessibility Tree	32.4	11.1	-
Vision-and-Language Modality				
Chart/ Document QA	Image	24.6	6.2	15.3
Non-GUI Perception	Image	28.7	7.6	14.0
GUI Perception	Image	27.4	7.1	14.0
Web Screenshot2Code	Image	28.0	6.6	9.9
Non-GUI Agents	Image	30.8	8.5	13.5
Multi-modal Math ✓	Image	30.4	8.5	15.3
Multi-round Visual Conversation	Image	30.0	9.0	12.6
Language Modality				
MathInstruct ✓	Image	31.9	10.9	14.4
Olympiad Math ✓	Image	31.5	8.5	13.1
CodeI/O ✓	Image	29.2	9.0	14.9
Web Knowledge Base	Image	31.3	9.5	9.0
Domain Combination (Sampled data from ✓domains)				
GUIMid	Image	34.3	9.5	21.2

Mid-Training

Domains	Observation	WebArena PR	WebArena SR	AndroidWorld SR
GUI Post-Training Only	Image	26.3	6.2	9.0
Public Baselines				
GPT-4o-2024-11-20	Image	36.9	15.6	11.7
OS-Genesis-7B	Image + Accessibility Tree	–	–	17.4
AGUVIS-72B	Image	-	-	26.1
Claude3-Haiku	Accessibility Tree	26.8	12.7	-
Llama3-70b	Accessibility Tree	35.6	12.6	-
Gemini1.5-Flash	Accessibility Tree	32.4	11.1	-
Vision-and-Language Modality				
Chart/ Document QA	Image	24.6	6.2	15.3
Non-GUI Perception	Image	28.7	7.6	14.0
GUI Perception	Image	27.4	7.1	14.0
Web Screenshot2Code	Image	28.0	6.6	9.9
Non-GUI Agents	Image	30.8	8.5	13.5
Multi-modal Math ✓	Image	30.4	8.5	15.3
Multi-round Visual Conversation	Image	30.0	9.0	12.6
Language Modality				
MathInstruct ✓	Image	31.9	10.9	14.4
Olympiad Math ✓	Image	31.5	8.5	13.1
CodeI/O ✓	Image	29.2	9.0	14.9
Web Knowledge Base	Image	31.3	9.5	9.0
Domain Combination (Sampled data from ✓domains)				
GUIMid	Image	34.3	9.5	21.2

Our 7B baselines achieve a comparable performance on AW, but relatively lower results on Web.

Mid-Training

Domains	Observation	WebArena PR	WebArena SR	AndroidWorld SR
GUI Post-Training Only	Image	26.3	6.2	9.0
Public Baselines				
GPT-4o-2024-11-20	Image	36.9	15.6	11.7
OS-Genesis-7B	Image + Accessibility Tree	–	–	17.4
AGUVIS-72B	Image	-	-	26.1
Claude3-Haiku	Accessibility Tree	26.8	12.7	-
Llama3-70b	Accessibility Tree	35.6	12.6	-
Gemini1.5-Flash	Accessibility Tree	32.4	11.1	-
Vision-and-Language Modality				
Chart/ Document QA	Image	24.6	6.2	15.3
Non-GUI Perception	Image	28.7	7.6	14.0
GUI Perception	Image	27.4	7.1	14.0
Web Screenshot2Code	Image	28.0	6.6	9.9
Non-GUI Agents	Image	30.8	8.5	13.5
Multi-modal Math ✓	Image	30.4	8.5	15.3
Multi-round Visual Conversation	Image	30.0	9.0	12.6
Language Modality				
MathInstruct ✓	Image	31.9	10.9	14.4
Olympiad Math ✓	Image	31.5	8.5	13.1
CodeI/O ✓	Image	29.2	9.0	14.9
Web Knowledge Base	Image	31.3	9.5	9.0
Domain Combination (Sampled data from ✓domains)				
GUIMid	Image	34.3	9.5	21.2

Generally, the similar domains (e.g. Document QA) do not help much on the Web, though they help some in the mobile tasks.

Mid-Training

Domains	Observation	WebArena		AndroidWorld
		PR	SR	SR
GUI Post-Training Only	Image	26.3	6.2	9.0
Public Baselines				
GPT-4o-2024-11-20	Image	36.9	15.6	11.7
OS-Genesis-7B	Image + Accessibility Tree	–	–	17.4
AGUVIS-72B	Image	-	-	26.1
Claude3-Haiku	Accessibility Tree	26.8	12.7	-
Llama3-70b	Accessibility Tree	35.6	12.6	-
Gemini1.5-Flash	Accessibility Tree	32.4	11.1	-
Vision-and-Language Modality				
Chart/ Document QA	Image	24.6	6.2	15.3
Non-GUI Perception	Image	28.7	7.6	14.0
GUI Perception	Image	27.4	7.1	14.0
Web Screenshot2Code	Image	28.0	6.6	9.9
Non-GUI Agents	Image	30.8	8.5	13.5
Multi-modal Math ✓	Image	30.4	8.5	15.3
Multi-round Visual Conversation	Image	30.0	9.0	12.6
Language Modality				
MathInstruct ✓	Image	31.9	10.9	14.4
Olympiad Math ✓	Image	31.5	8.5	13.1
CodeI/O ✓	Image	29.2	9.0	14.9
Web Knowledge Base	Image	31.3	9.5	9.0
Domain Combination (Sampled data from ✓domains)				
GUIMid	Image	34.3	9.5	21.2

All math-related domains help! Even the language math data, demonstrates generalization from text to multimodal tasks.

Mid-Training

Here we have some useful domains, what if we combine them?

We combine the math and code data and sample a 300K mid-training data: GUIMid

GUIMid

Domains	Observation	WebArena		AndroidWorld
		PR	SR	SR
GUI Post-Training Only	Image	26.3	6.2	9.0
Public Baselines				
GPT-4o-2024-11-20	Image	36.9	15.6	11.7
OS-Genesis-7B	Image + Accessibility Tree	–	–	17.4
AGUVIS-72B	Image	-	-	26.1
Claude3-Haiku	Accessibility Tree	26.8	12.7	-
Llama3-70b	Accessibility Tree	35.6	12.6	-
Gemini1.5-Flash	Accessibility Tree	32.4	11.1	-
Vision-and-Language Modality				
Chart/ Document QA	Image	24.6	6.2	15.3
Non-GUI Perception	Image	28.7	7.6	14.0
GUI Perception	Image	27.4	7.1	14.0
Web Screenshot2Code	Image	28.0	6.6	9.9
Non-GUI Agents	Image	30.8	8.5	13.5
Multi-modal Math ✓	Image	30.4	8.5	15.3
Multi-round Visual Conversation	Image	30.0	9.0	12.6
Language Modality				
MathInstruct ✓	Image	31.9	10.9	14.4
Olympiad Math ✓	Image	31.5	8.5	13.1
CodeI/O ✓	Image	29.2	9.0	14.9
Web Knowledge Base	Image	31.3	9.5	9.0
Domain Combination (Sampled data from ✓domains)				
GUIMid	Image	34.3	9.5	21.2

The combined data shows a significant improvement, especially on mobile, indicating these math and code data can complement each other, further enhancing the model's reasoning ability when combined.

Next Step:

We now have powerful agents capable of both planning and making action.

However, a single agent always has performance limits.

So …

How about bringing more agents to the party? 

AgentStore: Scalable Integration of Heterogeneous Agents As Specialized Generalist Computer Assistant

ACL 2025 VIENNA

Chengyou Jia, Minnan Luo, Zhuohang Dang, Qiushi Sun, Fangzhi Xu, Junlin Hu, Tianbao Xie, Zhiyong Wu

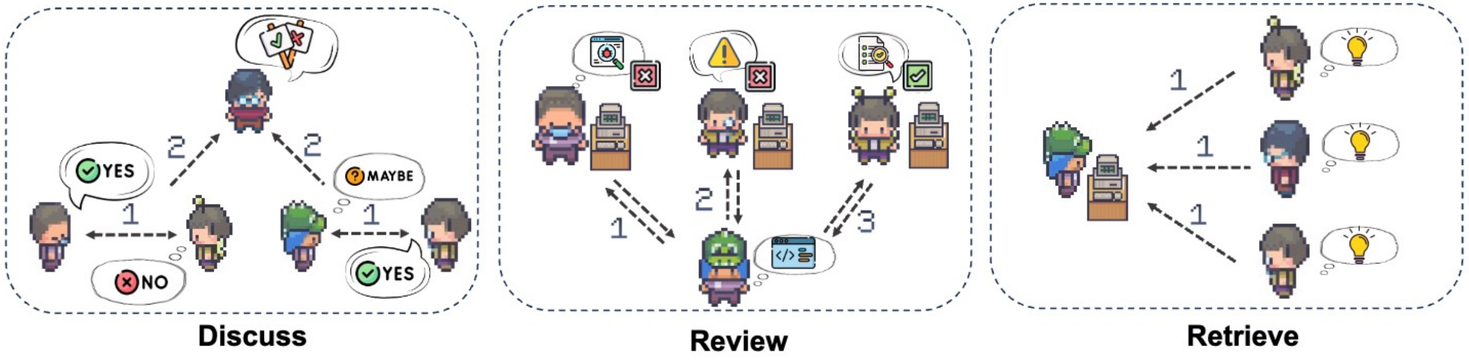
上海人工智能实验室
Shanghai Artificial Intelligence Laboratory

Multi-Agent Algorithms

Published as a conference paper at COLM 2024

Corex: Pushing the Boundaries of Complex Reasoning through Multi-Model Collaboration

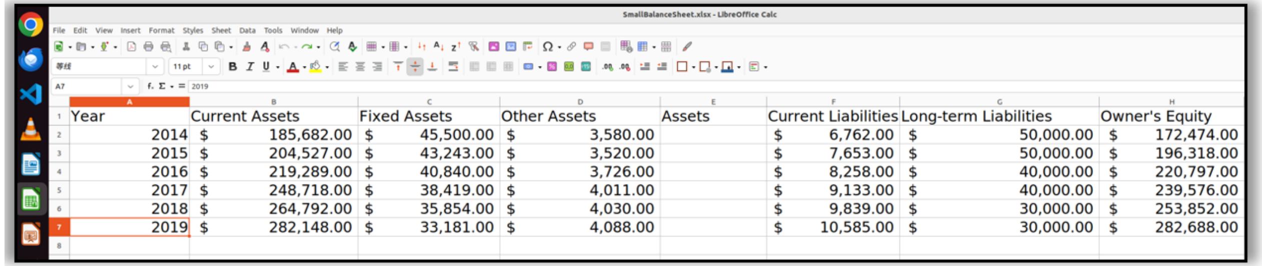
Qiushi Sun[◇♡*] **Zhangyue Yin**[♣] **Xiang Li**[♠] **Zhiyong Wu**[◇†] **Xipeng Qiu**[♣] **Lingpeng Kong**[♡]
[◇]Shanghai AI Laboratory [♡]The University of Hong Kong
[♣]Fudan University [♠]East China Normal University
qiushisun@connect.hku.hk, yinzy21@m.fudan.edu.cn, xiangli@dase.ecnu.edu.cn
wuzhiyong@pjlab.org.cn, xpqiu@fudan.edu.cn, lpk@cs.hku.hk



How about multi-agent + GUI Agents

Can a Single Agent handle a variety of OS tasks?

Task_1: In a new sheet with 4 headers "Year", "CA changes", "FA changes", and "OA changes", calculate the annual changes for the Current Assets, Fixed Assets, and Other Assets columns.



Year	Current Assets	Fixed Assets	Other Assets	Assets	Current Liabilities	Long-term Liabilities	Owner's Equity
2014	$ 185,682.00	$ 45,500.00	$ 3,580.00		$ 6,762.00	$ 50,000.00	$ 172,474.00
2015	$ 204,527.00	$ 43,243.00	$ 3,520.00		$ 7,653.00	$ 50,000.00	$ 196,318.00
2016	$ 219,289.00	$ 40,840.00	$ 3,726.00		$ 8,258.00	$ 40,000.00	$ 220,797.00
2017	$ 248,718.00	$ 38,419.00	$ 4,011.00		$ 9,133.00	$ 40,000.00	$ 239,576.00
2018	$ 264,792.00	$ 35,854.00	$ 4,030.00		$ 9,839.00	$ 30,000.00	$ 253,852.00
2019	$ 282,148.00	$ 33,181.00	$ 4,088.00		$ 10,585.00	$ 30,000.00	$ 282,688.00

SheetAgent
specialize in sheet processing

Step 1: Install and locate

pip install openpyxl && lsof | grep '.xlsx'

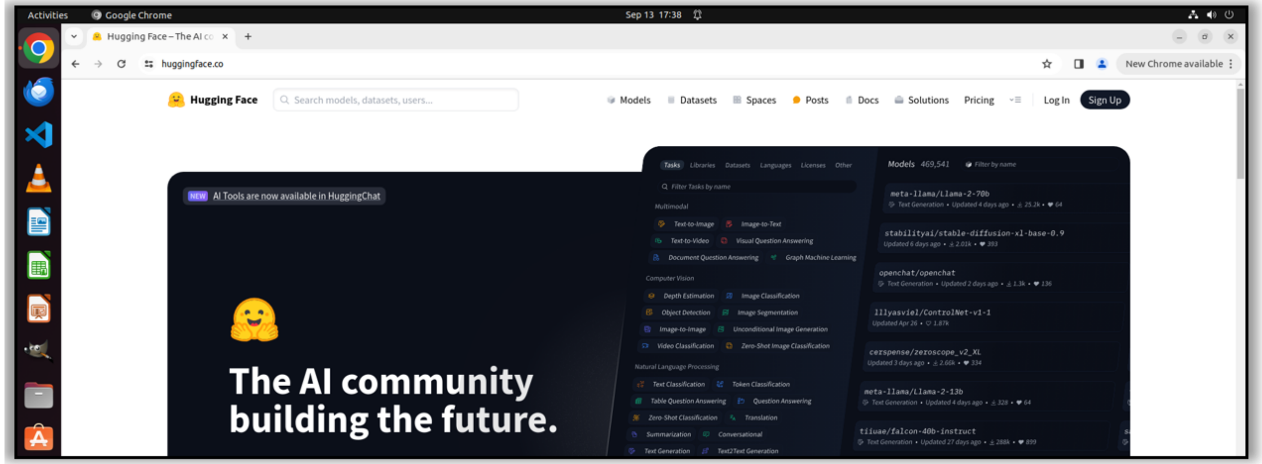
Step 2: Create new sheet and add headers

ws_new = wb.create_sheet(title=sheet_name)
ws_new.append(headers), wb.save(file_path)

Step 3: Insert table for the required data

for row in range(2, ws_original.max_row + 1):
year = ws_original.cell(arg).value,…
ws_new.append([year, …])

Task_2: Find the daily paper and take down the meta information of papers on 1st March, 2024 in the opened . pptx file. Please conform to the format and complete others.



WebAgent
specialize in web browsing

SildeAgent
specialize in slide editing

Different specialist agents are required to collaborate system-wide tasks

SubTask 1: Find papers and extract meta info

Step 1: Click daily papers to browsing
Step 2: Filter results by choosing1st March
Step 3: Extract info for selecting papers

subtask complete — message passing

SubTask 2: write meta info into pptx

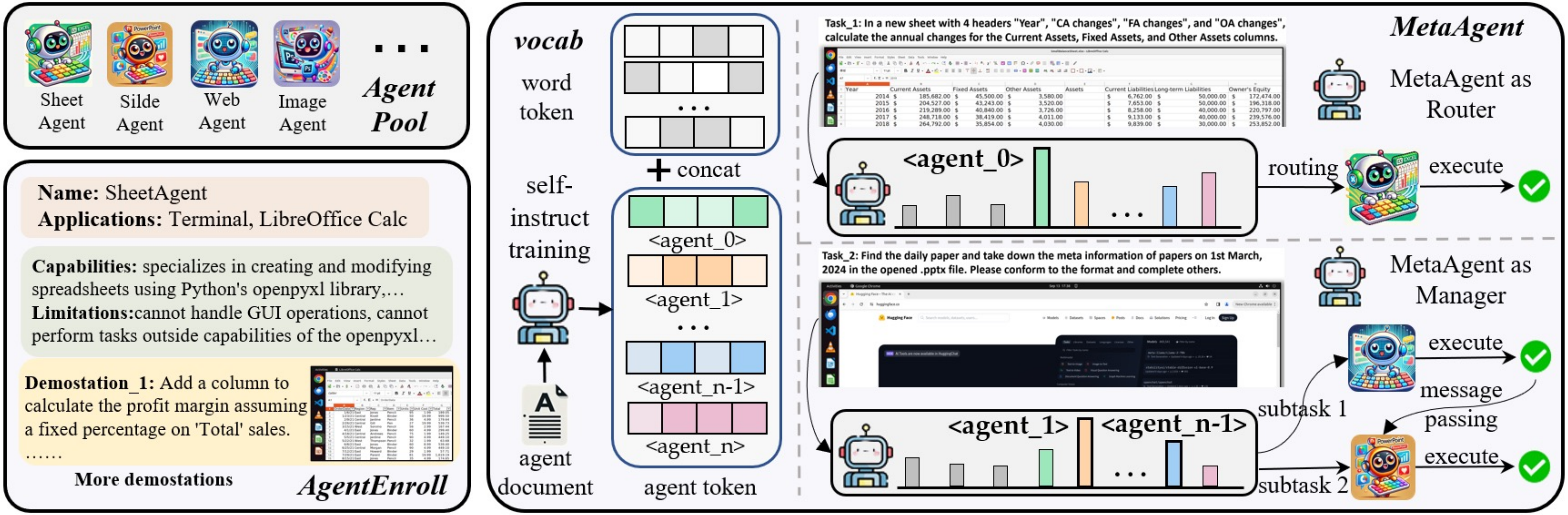
Step 1: Install package and locate .pptx file
Step 2: load content for current .pptx file
Step 3: Write info into corresponding file
Step 4: Save and overwrite the original file

1. Generalist Agent: lack of specialized abilities.
2. Specialized Agent: Unable to generalize to system-level tasks.

From APPStore to AgentStore:

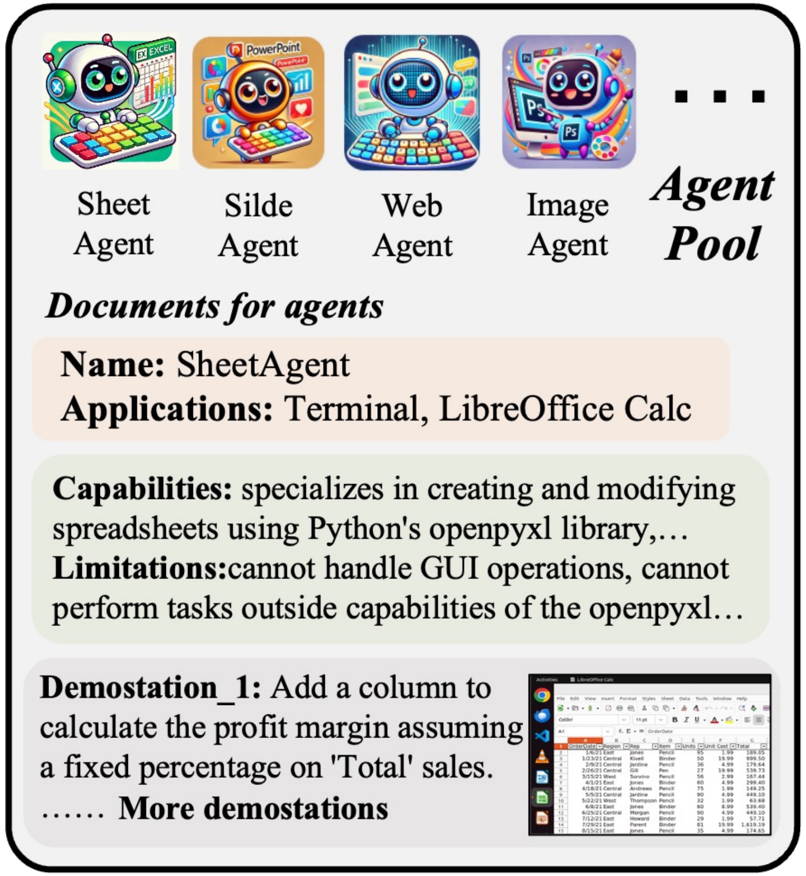
Build an open and scalable platform for dynamically integrating various computer-using agents.

AgentStore



1. AgentStore allows users to quickly integrate their own specialized agents into the platform, similar to the functionality of the App store.
2. We introduce a novel MLLM-based MetaAgent with AgentToken strategy, to select the most suitable agent(s) to complete tasks.

AgentStore



AgentPool: The set of all available agents in AgentStore.

1. Register new agents in a standardized format.
2. includes: functionality, limitations, application scenarios…
3. Define as $a = \{(a_1, d_1)(a_2, d_2), \ldots (a_n, d_n)\}$

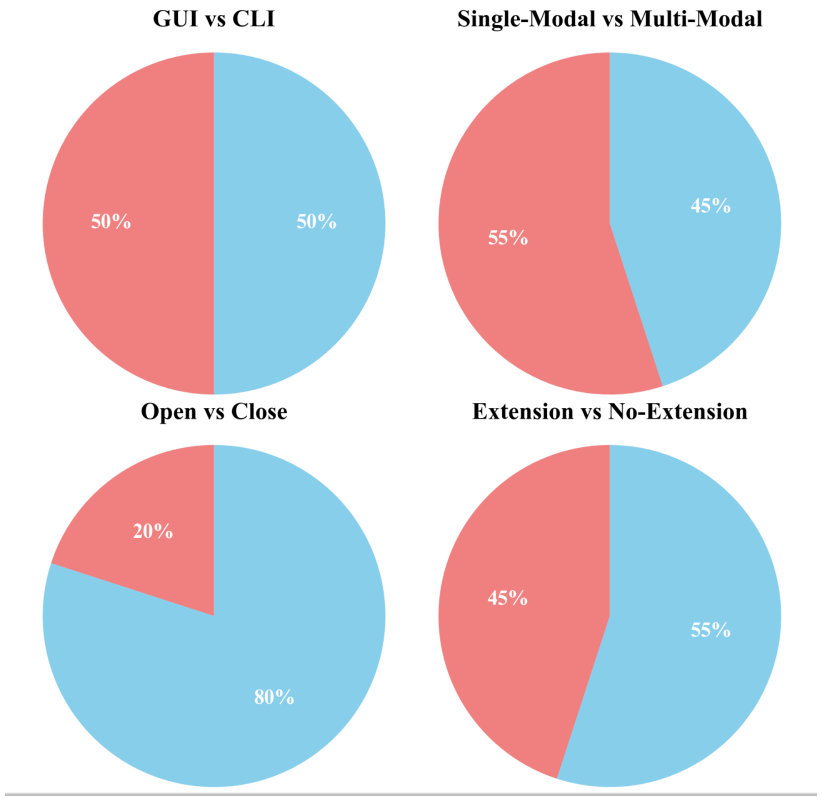
Agent   Documentation

20 desktop agents and 10 mobile agents, each specialized for tasks on their respective platforms.

Specialized agents in AgentStore

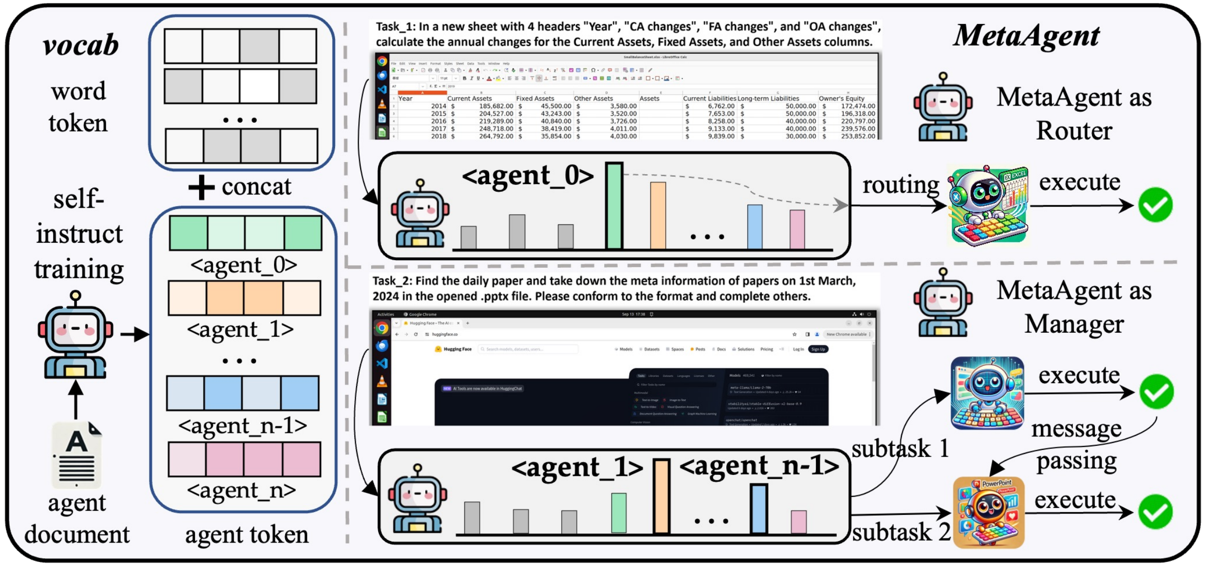
Table 6: The presentation of agents in the AgentPool.

	CLI or GUI?	Single or Multi Modal?	Open or Close Base Model?	Domain for OSworld	Support Extension?
OSAgent	GUI	Multi	Close	OS	✓
Friday (Wu et al., 2024)	CLI	Single	Close	OS	✓
SheetAgent	CLI	Single	Close	Calc	✗
CalcAgent	GUI	Multi	Close	Calc	✓
SlideAgent	CLI	Single	Close	Impress	✗
ImPressAgent	GUI	Multi	Close	Impress	✓
WordAgent	CLI	Single	Close	Writer	✗
WriterAgent	GUI	Multi	Close	Writer	✓
VLCAgent	GUI	Multi	Close	VLC	✓
MailAgent	GUI	Multi	Close	TB	✓
ChromeAgent	GUI	Multi	Close	Chrome	✓
WebAgent (He et al., 2024)	GUI	Multi	Close	Chrome	✗
VSAgent	GUI	Multi	Open	VSC	✗
VSGUIAgent	CLI	Single	Close	VSC	✓
GimpAgent	GUI	Multi	Close	GIMP	✓
ImageAgent	CLI	Single	Open	GIMP	✓
Searcher	CLI	Single	Close	-	✗
GoogleDrive	CLI	Single	Close	-	✗
CoderAgent	CLI	Single	Open	-	✗
VisionAgent	CLI	Multi	Open	-	✗



LLM/CLI-based model + LVM/GUI-based model

AgentStore



AgentToken: Each agent is registered by adding a token to the MetaAgent Vocab.

MetaAgent: Acts as an efficient router, predicting the most probable next token by maximizing conditional probability.

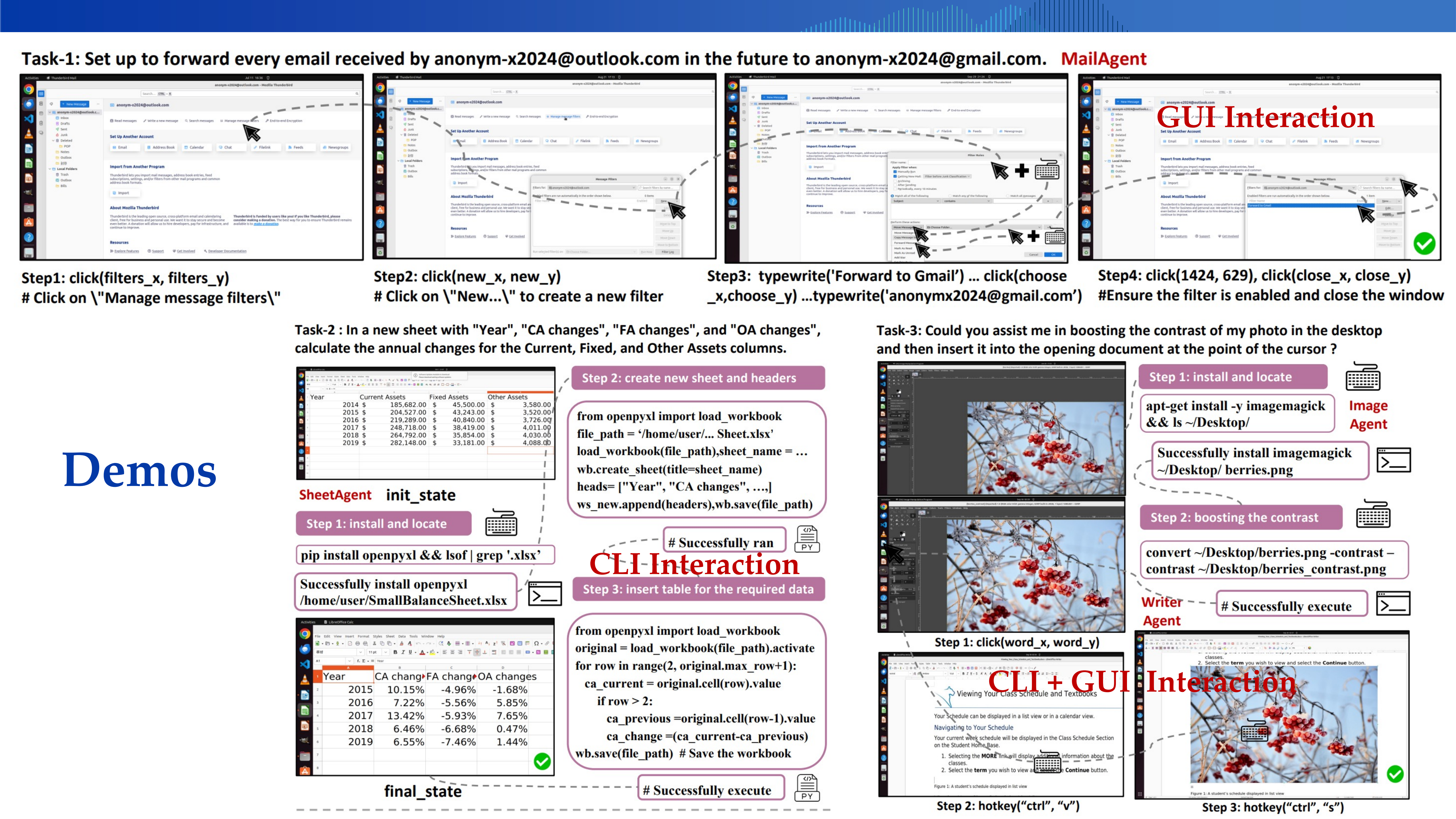
Once the agent token is predicted, decoding stops, and the corresponding Computer-using agent is called to execute the task.

Performance

Agent	Base	Success Rate (%)									
		OS*	Calc	Impress	Writer	VLC	TB	Chrome	VSC	GIMP	AVG
CogAgent	GogVLM	1.60	2.17	0.00	4.35	6.53	0.00	2.17	0.00	0.00	1.32
MMAgent	GPT-4o	14.44	4.26	6.81	8.70	9.50	6.67	15.22	30.43	0.00	11.21
CRADLE	GPT-4o	8.00	0.00	4.65	8.70	6.53	0.00	8.70	0.00	38.46	7.81
Friday*	GPT-4o	15.20	25.50	0.00	21.73	0.00	0.00	0.00	17.39	15.38	11.11
Open-Inter*	GPT-4o	12.80	12.76	0.00	13.04	0.00	0.00	0.00	17.39	15.38	8.94
AgentStore(GT)	Hybrid	20.00	36.17	10.63	47.83	47.06	40.00	34.78	47.82	38.46	29.54
AgentStore(ICL)	Hybrid	9.60	0.00	2.13	4.34	35.29	33.33	30.43	30.43	15.38	13.55
AgentStore(FT)	Hybrid	8.80	27.65	4.26	13.04	41.17	40.00	34.78	8.60	15.38	17.34
AgentStore(AT)	Hybrid	13.86	31.91	8.51	39.13	47.06	40.00	32.61	39.13	30.77	23.85

Rank	Model
1 Oct 24, 2024	AgentStore (AgentToken) *Shanghai AI Lab* Shanghai AI Lab, '24
2 Oct 11, 2024	Agent S w/ GPT-4o *Simular Research* Simular Research, '24
3 Oct 11, 2024	Agent S w/ Claude-3.5 *Simular Research* Simular Research, '24
4 Oct 24, 2024	AgentStore (Fine-Tuning) *Shanghai AI Lab* Shanghai AI Lab, '24
5 Oct 24, 2024	AgentStore (In-Context Learning) *Shanghai AI Lab* Shanghai AI Lab, '24
6 Mar 20, 2024	GPT-4 Vision *OpenAI* OpenAI, '23

AgentStore achieved a success rate of 23.85% on highly challenging OSWorld benchmark. (Claude 3.5 Sonnet: 22%)



Task-1: Set up to forward every email received by anonym-x2024@outlook.com in the future to anonym-x2024@gmail.com.
MailAgent
GUI Interaction
Step1: click(filters_x, filters_y)
Click on \"Manage message filters\"
Step2: click(new_x, new_y)
Click on \"New...\" to create a new filter
Step3: typewrite('Forward to Gmail') ... click(choose_x,choose_y) ...typewrite('anonymx2024@gmail.com')
Step4: click(1424, 629), click(close_x, close_y)
#Ensure the filter is enabled and close the window
Demos
Task-2 : In a new sheet with "Year", "CA changes", "FA changes", and "OA changes", calculate the annual changes for the Current, Fixed, and Other Assets columns.
Year
Current Assets
Fixed Assets
Other Assets
2014 $ 185,682.00 $ 45,500.00 $ 3,580.00
2015 $ 204,527.00 $ 43,243.00 $ 3,520.00
2016 $ 219,289.00 $ 40,840.00 $ 3,726.00
2017 $ 248,718.00 $ 38,419.00 $ 4,011.00
2018 $ 264,792.00 $ 35,854.00 $ 4,030.00
2019 $ 282,148.00 $ 33,181.00 $ 4,088.00
SheetAgent
init_state
Step 1: install and locate
pip install openpyxl && lsof | grep '.xlsx'
Successfully install openpyxl
/home/user/SmallBalanceSheet.xlsx
Step 2: create new sheet and headers
from openpyxl import load_workbook
file_path = '/home/user/... Sheet.xlsx'
load_workbook(file_path),sheet_name = ...
wb.create_sheet(title=sheet_name)
heads= ["Year", "CA changes", ...,]
ws_new.append(headers),wb.save(file_path)
# Successfully ran
PY
CLI Interaction
Step 3: insert table for the required data
from openpyxl import load_workbook
original = load_workbook(file_path).activate
for row in range(2, original.max_row+1):
ca_current = original.cell(row).value
if row > 2:
ca_previous =original.cell(row-1).value
ca_change =(ca_current-ca_previous)
wb.save(file_path) # Save the workbook
Successfully execute
Year
CA chang
FA chang
OA changes
2015 10.15% -4.96% -1.68%
2016 7.22% -5.56% 5.85%
2017 13.42% -5.93% 7.65%
2018 6.46% -6.68% 0.47%
2019 6.55% -7.46% 1.44%
final_state
Task-3: Could you assist me in boosting the contrast of my photo in the desktop and then insert it into the opening document at the point of the cursor ?
Step 1: install and locate
apt-get install -y imagemagick && ls ~/Desktop/
Image Agent
Successfully install imagemagick
~/Desktop/ berries.png
Step 2: boosting the contrast
convert ~/Desktop/berries.png -contrast –contrast ~/Desktop/berries_contrast.png
Writer Agent
Successfully execute
Step 1: click(word_x, word_y)
CLI + GUI Interaction
Viewing Your Class Schedule and Textbooks
Your Schedule can be displayed in a list view or in a calendar view.
Navigating to Your Schedule
Your current week schedule will be displayed in the Class Schedule Section on the Student Home Base.
Step 2: hotkey("ctrl", "v")
Step 3: hotkey("ctrl", "s")

Summary of Multi-Agents

1. Multi-agent integration can rapidly advance computer-using capabilities.
2. Greatly facilitates generalization to new domains.
3. Plug-and-play design, enabled by carefully crafted AgentTokens, allows for fast integration.

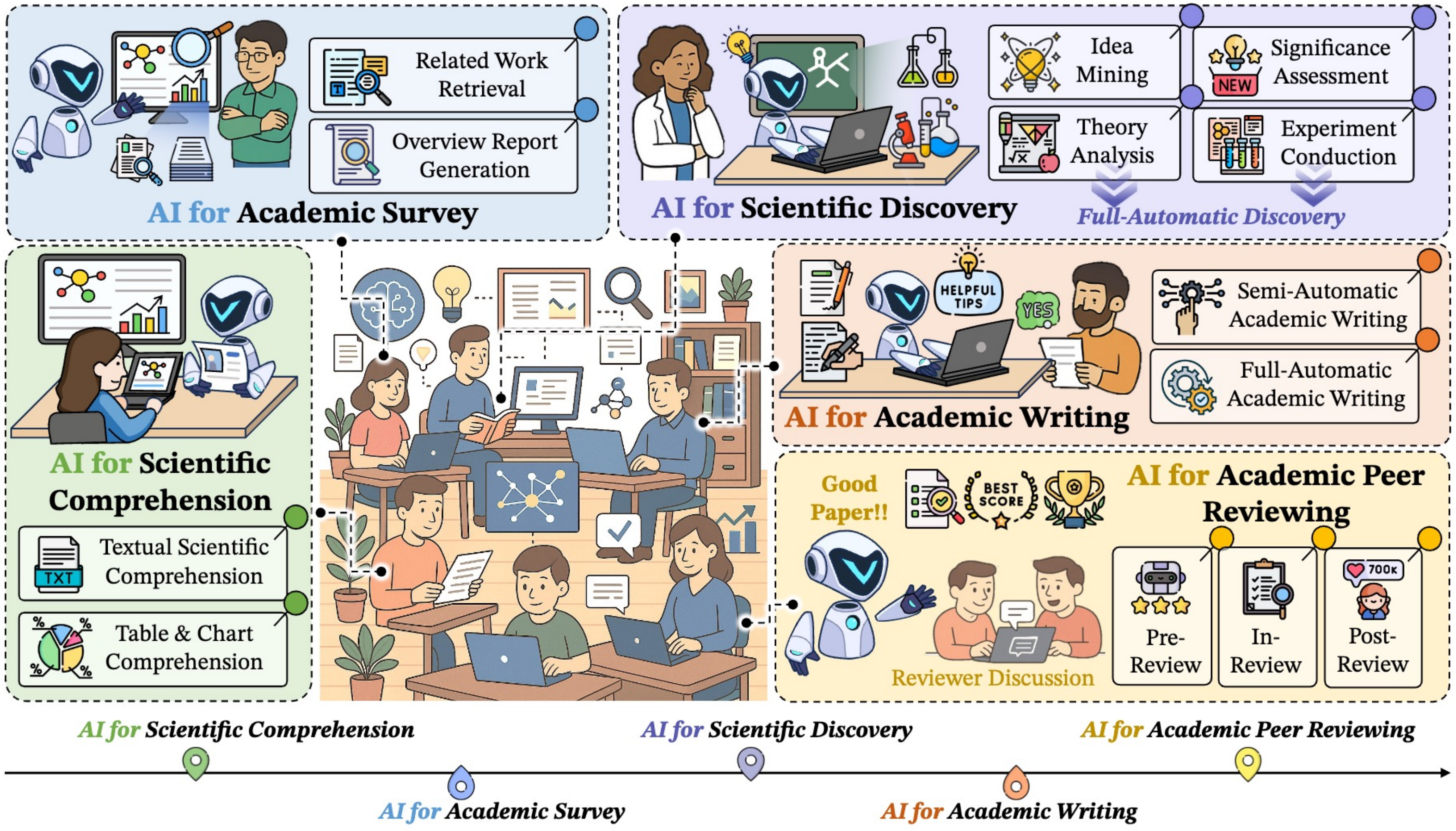
中文解读（AgentStore）

Next Steps?

Exploring the deep value of computer-using agents: from general-purpose scenarios to specialized professional applications.

Backgrounds

AI4Research is a highly popular concept.



[17] *AI4Research: A Survey of Artificial Intelligence for Scientific Research*

Backgrounds: Pastoral Age

Fig. 4 Most frequent topics in the BioASQ questions.

BioASQ-QA (Nature 2023)

- Designed for biomedical question answering
- Annually expanded with new questions and answers.
- Available on Zenodo in JSON format.

MoleculeQA (ArXiv 2024)

- Evaluate Factual Accuracy in Molecular Comprehension
- 62K QA Pairs across 23K molecules
- MCQ problems (training set available)
- Textual-based

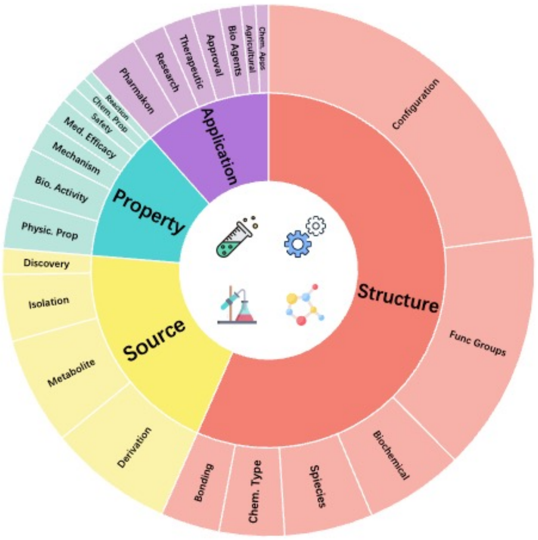


Figure 4: An overview of MoleculeQA topics distribution. Four coarse-grained aspects occupy the inner circle, and in the outer circle we list finer-grained non-leaf topics.

[18] BioASQ-QA: A manually curated corpus for Biomedical Question Answering, Krithara et al, Nature 2023

[19] MoleculeQA: A Dataset to Evaluate Factual Accuracy in Molecular Comprehension, Lu, et al, ArXiv 2024

Backgrounds: Contemporary Era

A lot of “AI Research” systems have been built…

2024-9-4

The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery

Chris Lu[1,2,*], Cong Lu[3,4,*], Robert Tjarko Lange[1,*], Jakob Foerster[2,†], Jeff Clune[3,4,5,†] and David Ha[1,†]
[*]Equal Contribution, [1]Sakana AI, [2]FLAIR, University of Oxford, [3]University of British Columbia, [4]Vector Institute, [5]Canada CIFAR AI Chair, [†]Equal Advising

SciMON 🧪 : Scientific Inspiration Machines Optimized for Novelty

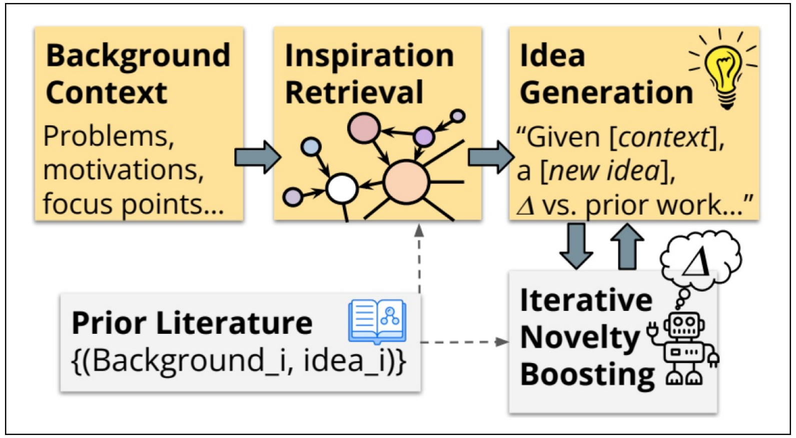
Qingyun Wang[1], Doug Downey[2], Heng Ji[1], Tom Hope[2,3]
[1] University of Illinois at Urbana-Champaign [2] Allen Institute for Artificial Intelligence (AI2)
[3] The Hebrew University of Jerusalem
{tomh,doug}@allenai.org, {qingyun4,hengji}@illinois.edu

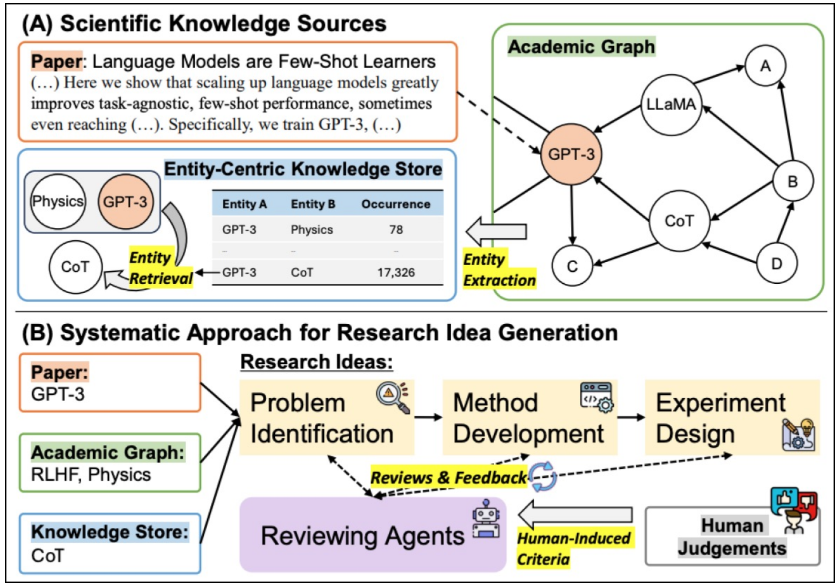
ResearchAgent: Iterative Research Idea Generation over Scientific Literature with Large Language Models

Jinheon Baek[1] Sujay Kumar Jauhar[2] Silviu Cucerzan[2] Sung Ju Hwang[1,3]
KAIST[1] Microsoft Research[2] DeepAuto.ai[3]
{jinheon.baek, sjhwang82}@kaist.ac.kr {sjauhar, silviu}@microsoft.com

Automated Peer Reviewing in Paper SEA: Standardization, Evaluation, and Analysis

Jianxiang Yu[♡*], Zichen Ding[♡*], Jiaqi Tan[♡], Kangyang Luo[♡], Zhenmin Weng[♡], Chenghua Gong[♡], Long Zeng[♡], Renjing Cui[♡], Chengcheng Han[♡], Qiushi Sun[♢], Zhiyong Wu[♢], Yunshi Lan[♡], Xiang Li[♡†]
[♡] East China Normal University, Shanghai, China
[♢] Shanghai AI Laboratory, Shanghai, China
sea.ecnu@gmail.com
https://ecnu-sea.github.io/





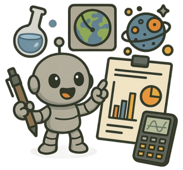
[20] The Dream of Automating Research, Stanford NLP

Thinking

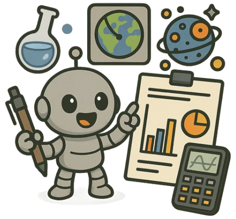
Traditionally, AI acted as an "analyzer," helping with idea thinking data analysis, writing, and visualization.

With Computer-using agents, AI can be evolved into an "executor" capable of directly operating scientific software via GUI or CLI,

Moving beyond QA to actively performing research tasks!



From Digital Agents to AI Co-Scientists



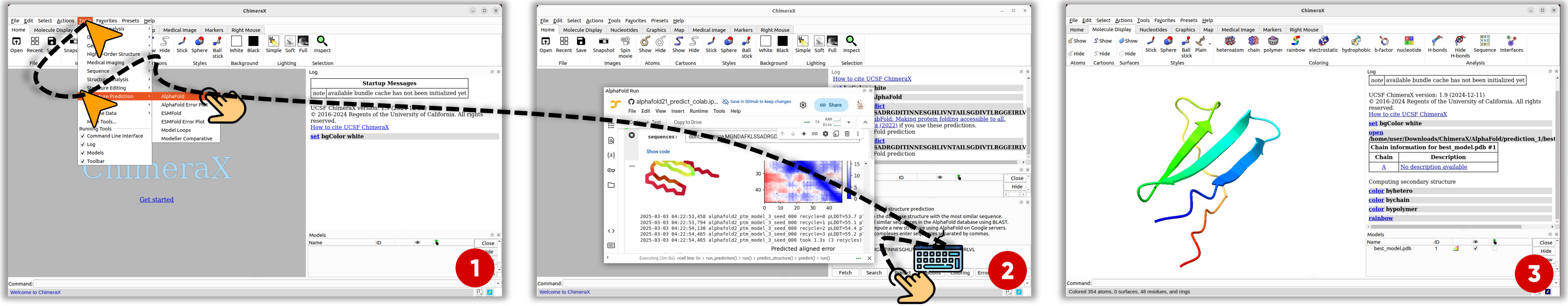
ScienceBoard: Evaluating Multimodal Autonomous Agents in Realistic Scientific Workflows

Qiushi Sun, Zhoumianze Liu, Chang Ma, Zichen Ding, Fangzhi Xu, Zhangyue Yin, Haiteng Zhao, Zhenyu Wu, Kanzhi Cheng, Zhaoyang Liu, Jianing Wang, Qintong Li, Xiangru Tang, Tianbao Xie, Xiachong Feng, Xiang Li, Ben Kao, Wenhai Wang, Biqing Qi, Lingpeng Kong, Zhiyong Wu

Preprint / WUCA @ ICML 2025 Oral

Use Cases

Instruction: Predict the protein structure for the amino acid sequence of 'MGND...' via AlphaFold in ChimeraX.



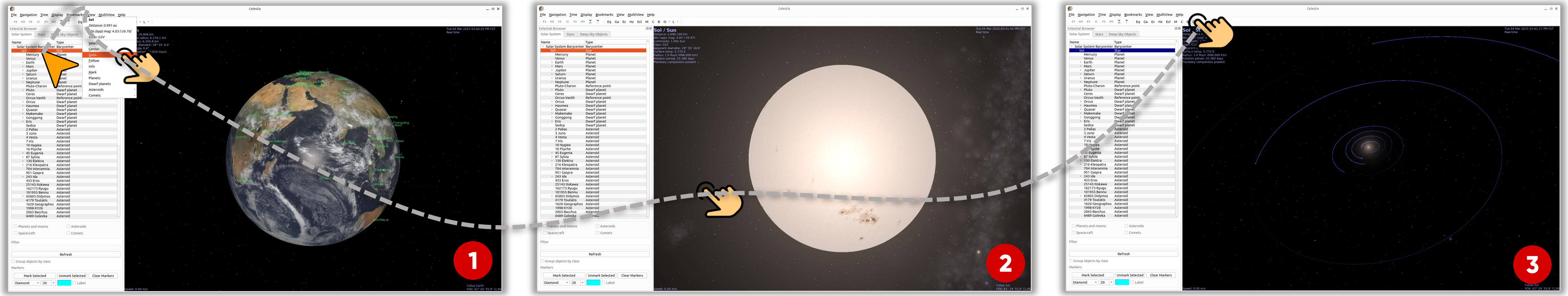
Step1: Toggle the widget of AlphaFold.

Step2: Input the given sequence and call out AlphaFold for structure prediction.

Step3: Wait until the prediction finished.

Use Cases

Instruction: Show planets' orbits of Solar System in Celestia.



Step1: Select the Sol and click 'Goto' in contect menu.

Step2: Slide the mouse wheel to move the camera away from Sol.

Step3: Click to show orbits of planets.

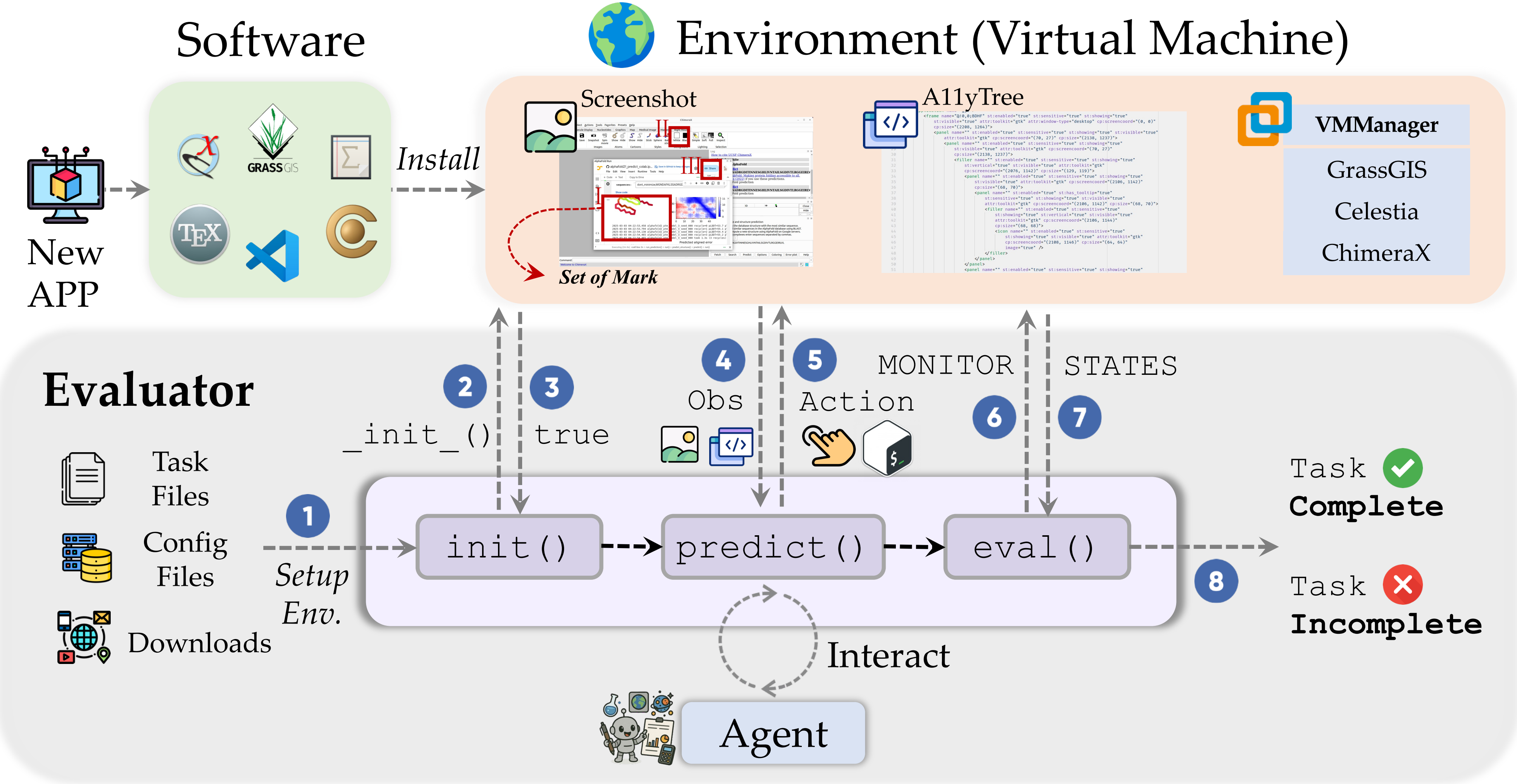
ScienceBoard

To reach such automation, a playground integrating

1. Scientific software
2. Evaluators

Is essential, a highly non-trivial endeavor!

ScienceBoard Infra



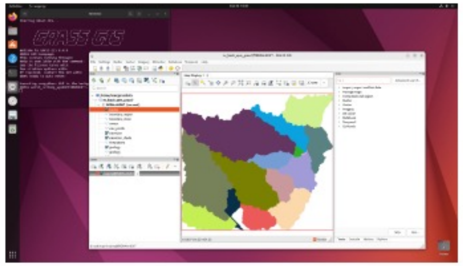
The first multimodal agent evaluation environment designed
for scientific tasks, real interactions, and automatic assessment

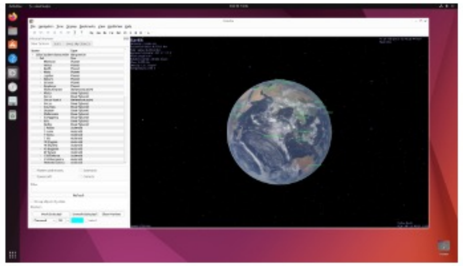
ScienceBoard Evaluation

State-based evaluation

Initial State	Instruction	Evaluation Script (Simplified)

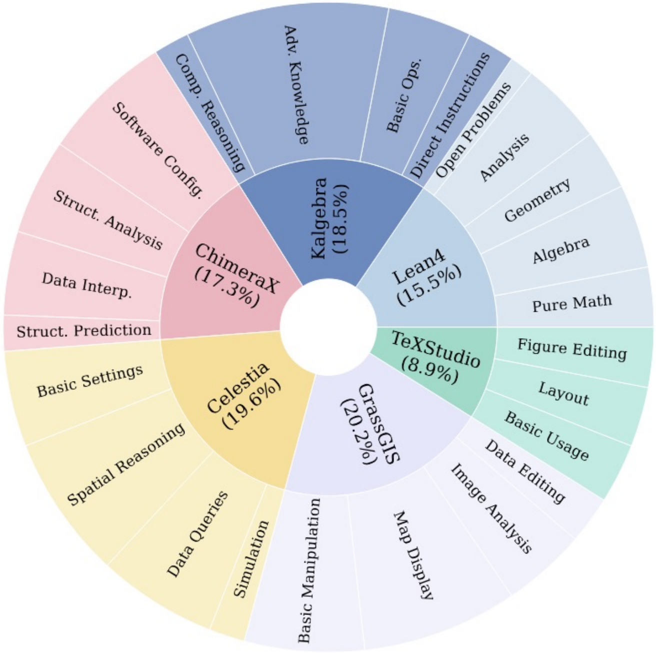
	*Select all water molecules and draw their centroids with radius of 1Å in ChimeraX.*	{
		"type":"info","key":"sell",
		"value":["atom id #!1/A:201@O idatm_type O3"
		"...",]
		},{
		"type":"states",
		"find":"lambda k,v:k.endswith('._name')",
		"key":"lambda k:'..._atoms_drawing'",
		"value":"[[13.0012 1.7766 21.3672 1.]]"
		}

	*Display and ONLY display the layer of 'boundary_region' in Grass GIS.*	{
		"type":"info",
		"key":"lambda dump:len(dump['layers'])",
		"value":1
		},{"type":"info"
		"key":"lambda dump:dump['layers'][0]['name']",
		"value":"boundary_region@PERMANENT"
		}

	*Set the Julian date to 2400000 in Celestia.*	{
		"type":"info",
		"key":"simTime",
		"value":2400000,
		"pred":"lambda left, right:abs(left-right) < 1",
		}

ScienceBoard Benchmark

Task Type	Statistics
Total Tasks	**169 (100%)**
- GUI	38 (22.5%)
- CLI	33 (19.5%)
- GUI + CLI	98 (58.0%)
Difficulty	
- Easy	91 (53.8%)
- Medium	48 (28.4%)
- Hard	28 (16.6%)
- Open Problems	2 (1.2%)
Instructions	
Avg. Length of Task Instructions	20.0
Avg. Length of Agentic Prompt	374.9
Execution	
Avg. Steps	9.0
Avg. Time Consumption	124(s)



Evaluate autonomous computer-using agents in realistic scientific workflows.

Tasks require complex tool usage, scientific reasoning, and multi-step GUI/CLI operations

Evaluation

Proprietary Models

Opensource LLM / VLMs

GUI Action Models

[21] *Navigating the Digital World as Humans Do: Universal Visual Grounding for GUI Agents*

[22] *UI-TARS: Pioneering Automated GUI Interaction with Native Agents*

[23] *GUI-Actor: Coordinate-Free Visual Grounding for GUI Agents*

Evaluation: General Setting

Overall success rate remains low (avg. ~15%)

Performance varies among domains

Best results achieved with combined Screenshot + a11ytree setting

Table 3: Success rates on ScienceBoard. We present the performance of each agent backbone across different scientific domains under various observation settings. Proprietary Models, Open-Source VLMs / LLMs, and GUI Action Model are distinguished by color.

Observations	Model	Success Rate (↑) Algebra	Biochem	GIS	ATP	Astron	Doc	Overall
Screenshot	GPT-4o	3.23%	0.00%	0.00%	0.00%	0.00%	6.25%	1.58%
	Claude-3.7-Sonnet	9.67%	37.93%	2.94%	0.00%	6.06%	6.25%	10.48%
	Gemini-2.0-Flash	6.45%	3.45%	2.94%	0.00%	0.00%	6.06%	3.15%
	Qwen2.5-VL-72B	22.58%	27.59%	5.88%	0.00%	9.09%	12.50%	12.94%
	InternVL3-78B	6.45%	3.45%	0.00%	0.00%	0.00%	6.25%	2.69%
	UI-TARS-1.5-7B	12.90%	13.79%	0.00%	0.00%	6.06%	0.00%	2.69%
a11ytree	GPT-4o	12.90%	20.69%	2.94%	0.00%	6.06%	0.00%	7.10%
	Claude-3.7-Sonnet	19.35%	34.48%	2.94%	3.85%	12.12%	0.00%	12.12%
	Gemini-2.0-Flash	9.68%	17.24%	0.00%	0.00%	0.00%	0.00%	4.49%
	o3-mini	16.13%	20.69%	2.94%	3.85%	15.15%	6.25%	10.84%
	Qwen2.5-VL-72B	9.68%	10.34%	2.94%	0.00%	3.03%	0.00%	4.33%
	InternVL3-78B	3.23%	3.45%	0.00%	0.00%	0.00%	0.00%	1.11%
Screenshot + a11ytree	GPT-4o	22.58%	37.93%	2.94%	7.69%	3.03%	12.50%	14.45%
	Claude-3.7-Sonnet	12.90%	41.37%	8.82%	3.85%	9.09%	18.75%	15.79%
	Gemini-2.0-Flash	16.13%	24.14%	2.94%	0.00%	18.18%	12.50%	12.32%
	Qwen2.5-VL-72B	16.13%	20.69%	2.94%	0.00%	18.18%	12.50%	11.74%
	InternVL3-78B	6.45%	3.45%	0.00%	0.00%	3.03%	6.25%	3.20%
Set-of-Mark	GPT-4o	6.45%	3.45%	0.00%	0.00%	3.03%	12.50%	4.24%
	Claude-3.7-Sonnet	16.13%	31.03%	5.88%	0.00%	6.06%	12.50%	11.93%
	Gemini-2.0-Flash	3.23%	0.00%	0.00%	0.00%	3.03%	6.25%	2.09%
	Qwen2.5-VL-72B	6.45%	6.90%	2.94%	0.00%	3.03%	12.50%	6.36%
	QvQ-72B-Preview	0.00%	0.00%	2.94%	0.00%	3.03%	0.00%	0.49%
	InternVL3-78B	3.23%	6.90%	2.94%	0.00%	0.00%	0.00%	2.18%
Human Performance		74.19%	68.97%	55.88%	42.31%	51.52%	68.75%	60.27%

Evaluation: Modular Setting

GPT-4o as the planner + GUI model

Clear performance improvement (up to ~20% SR)

Separating planning and action offers a promising direction!

Table 4: Success rates of different VLM agent combinations under the planner + grounding model setting on SCIENCEBOARD. The observation setting used in this experiment is screenshot. Colors denote Proprietary Models, Open-Source VLMs and GUI Action Models.

Planner	Grounding Model	Success Rate (↑)				
		Algebra	Biochem	GIS	Astron	Overall
GPT-4o	OS-Atlas-Pro-7B	6.25%	10.34%	0.00%	3.03%	4.92%
	UGround-V1-7B	0.00%	3.45%	0.00%	3.03%	1.62%
	Qwen2.5-VL-72B	12.50%	34.48%	11.76%	9.09%	16.96%
	UI-TARS-72B	3.23%	10.34%	5.88%	6.06%	6.38%
	GUI-Actor-7B	21.88%	44.83%	2.94%	12.12%	20.44%
GPT-4o		3.23%	0.00%	0.00%	0.00%	0.81%

Next step: stronger multi-agent system + domain knowledge?

Leaderboard

Screenshot | A11y Tree | Screenshot + A11y Tree | Set of Marks | Search by keywords

O..	Settings	% Acc	% Alg	% Biochem	% GIS	% ATP	% Astron	% Doc
	Calude-3.7-Sonnet w/ screenshot…	15.79	12.90	41.37	8.82	3.85	9.09	18.75
	GPT-4o (2024-08-06) w/ screensh…	14.45	22.58	37.93	2.94	7.69	3.03	12.50
	GPT-4o (2024-08-06) w/ set_of_m…	14.45	6.45	3.45	0.00	0.00	3.03	12.50
	Qwen2.5-VL-72B w/ screenshot	12.94	22.58	27.59	5.88	0.00	9.09	12.50
	Gemini-2.0-Flash w/ screenshot+a…	12.32	16.13	24.14	2.94	0.00	18.18	12.50
	Calude-3.7-Sonnet w/ a11y_tree	12.12	19.35	34.48	2.94	3.85	12.12	0.00
	Calude-3.7-Sonnet w/ set_of_marks	11.93	16.13	31.03	5.88	0.00	6.06	12.50
	Qwen2.5-VL-72B w/ screenshot+a…	11.74	16.13	20.69	2.94	0.00	18.18	12.50
	o3-mini (2025-01-31) w/ a11y_tree	10.84	16.13	20.69	2.94	3.85	15.15	6.25
	Calude-3.7-Sonnet w/ screenshot	10.48	9.67	37.93	2.94	0.00	6.06	6.25
	GPT-4o (2024-08-06) w/ a11y_tree	7.10	12.90	20.69	2.94	0.00	0.00	6.06
	Qwen2.5-VL-72B w/ set_of_marks	6.36	6.45	6.90	2.94	0.00	3.03	12.50
	UI-TARS-1.5 w/ screenshot	5.92	12.90	13.79	0.00	0.00	6.06	0.00
	Gemini-2.0-Flash w/ a11y_tree	4.49	9.68	17.24	0.00	0.00	0.00	0.00
	Qwen2.5-VL-72B w/ a11y_tree	4.33	9.68	10.34	2.94	0.00	3.03	0.00
	InternVL3-78B w/ screenshot+a11…	3.20	6.45	3.45	0.00	0.00	3.03	6.25
	Gemini-2.0-Flash w/ screenshot	3.15	6.45	3.45	2.94	0.00	0.00	6.06

https://qiushisun.github.io/ScienceBoard-Home/

Our Project

ScienceBoard

Evaluating Multimodal Autonomous Agents in Realistic Scientific Workflows

Introducing ScienceBoard, a first-of-its-kind evaluation platform for multimodal agents in ***scientific workflows***. ScienceBoard is characterized by the following core features:

- **Pioneering Application**: ScienceBoard is the first to bring computer-using agents into the domain of scientific discovery, enabling autonomous research assistants across disciplines.
- **Realistic Environment**: We provide a dynamic, visually grounded virtual environment integrated with professional scientific software, supporting both GUI and CLI interaction in real-time workflows.
- **Challenging Benchmark**: A new benchmark of 169 rigorously validated tasks across 6 core domains is introduced, capturing real-world challenges.
- **Comprehensive Evaluations**: We presents systematic evaluations across a wide range of agents powered by LLMs, VLMs, and GUI action models.

arXiv | Code | Data | VM Snapshot

中文解读（ScienceBoard）

https://qiushisun.github.io/ScienceBoard-Home/

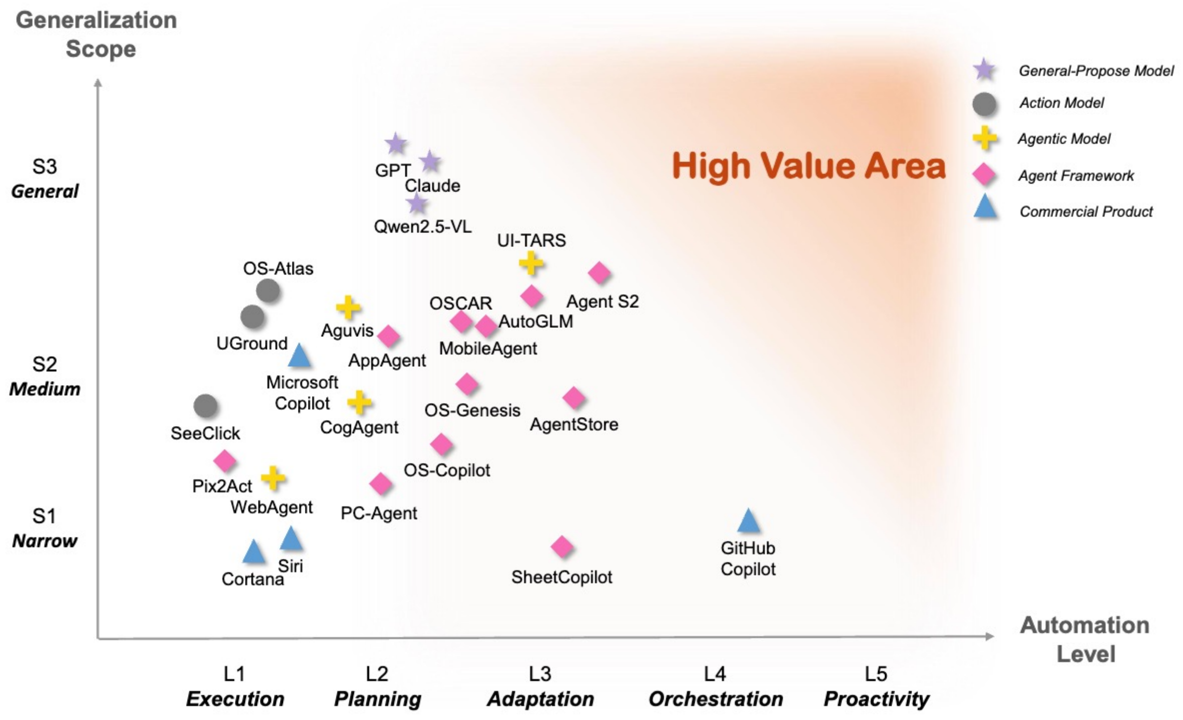
Future

We are just standing at the dawn of a long journey!

1. Holistic Evaluation? 
2. Agent Safety? 
3. Efficiency? 
4. Physical world? 
5. …

Holistic Evaluation

The development of computer-using agents has been rapidly advancing, yet systematic evaluation remains underexplored.



Stay tuned!

OS-MAP: How Far Can Computer Use Agents Go in Breadth and Depth?

Preprint / WUCA @ ICML International Conference On Machine Learning 2025

Safety Concerns

Agent safety research is behind agent deployment!

Efficiency

Although computer-using agents can accomplish many tasks, **efficiency remains a critical concern**.

Two main aspects:

1. Training efficiency: Heavy reliance on massive data (data hungry)
2. Inference efficiency: High latency during real-time execution

Connection to the Physical World

How can computer-using agents achieve embodiment?

1. Robotic arms?
2. Exoskeletons?
3. …

Future

We are just standing at the dawn of a long journey!

1. Holistic Evaluation?
2. Agent Safety?
3. Efficiency?
4. Physical world?
5. …

中文解读（OS-Genesis）

中文解读（ScienceBoard）

中文解读（SeeClick）

中文解读（OS-ATLAS）

中文解读（AgentStore）

Acknowledgement

We are just standing at the dawn of a long journey!

上海人工智能实验室
Shanghai Artificial Intelligence Laboratory

JOHNS HOPKINS UNIVERSITY

CAMEL-AI

Thanks for listening

Contact: qiushisun@connect.hku.hk