



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



SCHOOL OF
COMPUTING &
DATA SCIENCE
The University of Hong Kong

Building GUI Agent Data with OS-Genesis

Qiushi Sun

qiushisun.github.io

✕ @qiushi_sun

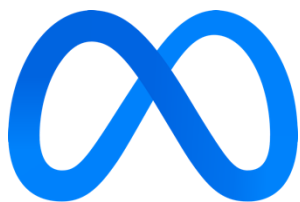
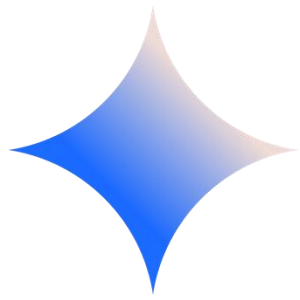
Computer Use Agents



The Feasibility of Jarvis AI from Marvel in Real Life

Computer Use Agents

Once out of reach, but we are turning it into reality.

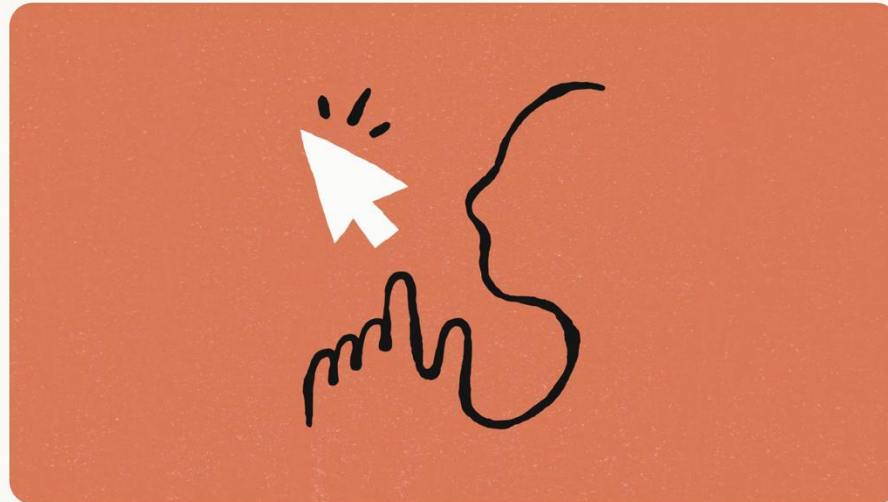


Computer Use Agents

Both academia and industry are building **computer use agents**

Introducing computer use, a new
Claude 3.5 Sonnet, and Claude 3.5
Haiku

22 Oct 2024 • 5 min read



Claude Computer Use

Computer Use Agents

Both academia and industry are building **computer use agents**

Introducing Operator

A research preview of an agent that can use its own browser to perform tasks for you. Available to Pro users in the U.S.

[Go to Operator ↗](#)

Find and book me the highest rated one-day tour of Rome on Tripadvisor.

I'll search for the highest-rated tour of historic Rome on TripAdvisor. Once I find a suitable option, I'll provide you with the details. Let's begin.

Worked for 2 minutes ^

- Navigating to TripAdvisor website
- Selecting "Things to Do" category
- Searching for historic Rome tours
- Closing pop-up, continuing tour search
- Searching for Historic Rome tours
- Exploring all historic Rome tour options
- Closing Colosseum tab, resuming tour search
- Closing tour pop-up, tab afterward
- Exploring options for top-rated tours
- Sorting results by tour ratings
- Exploring filters for top-rated tours
- Scrolling for sorting options, finding tours

Tripadvisor

Rome: Colosseum, Roman Forum and Palatine Hill

By CityWonders

5,220 reviews

OpenAI Operator

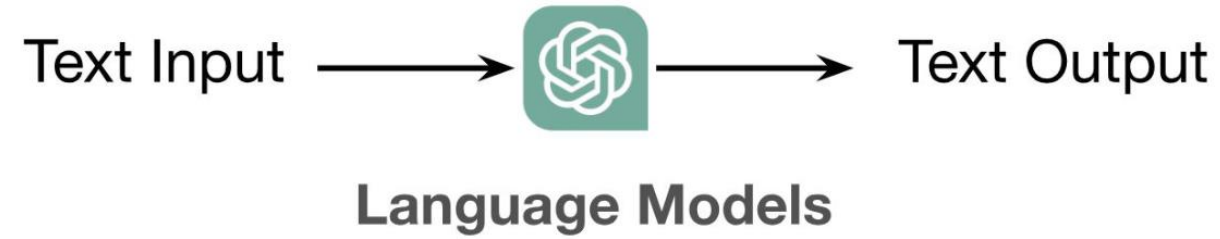
Computer Use Agents

They are quite promising for achieving **Digital Automation**.

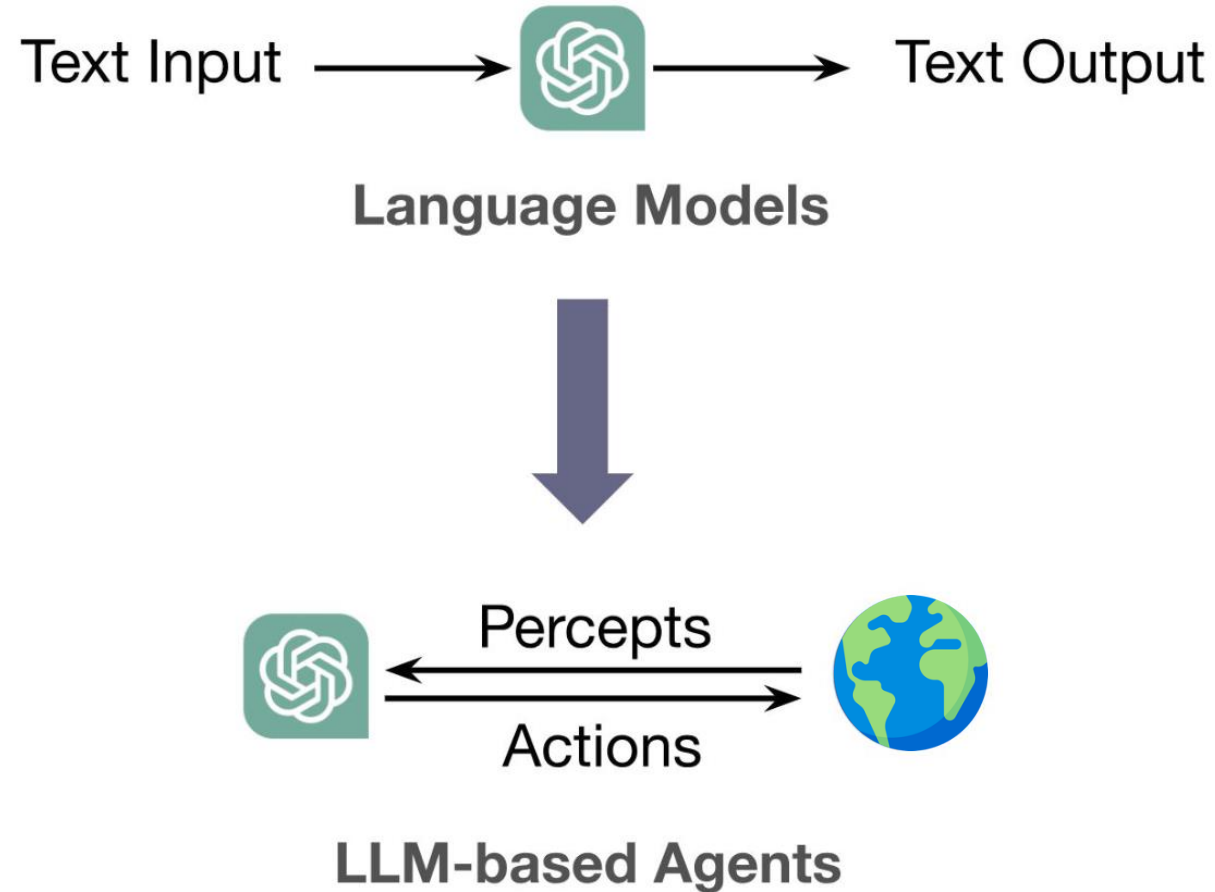
Can we transform a (V)LM into such **computer / GUI agents**?

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

Recap: Language Agents






Recap: Language Agents



But this is not enough for Computer Use / GUI Agents.

Computer Use 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 Use Agents

Behavioral Cloning / Imitation Learning.



Sounds good, but where is our **data**?

Data Scarcity

Data curation is **much more expensive** than you think. 

Take Scale AI as an example.

Not to mention scenario/domain - specific data.



Alexandr Wang   @alexandr_wang · Jan 24

An interview today where I talk about how it relates to the US/China race and DeepSeek's score:



From cnbc.com

 48

 87

 279

 56K



Data Scarcity

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

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



GUI Trajectory Data

The best data format of GUI agents

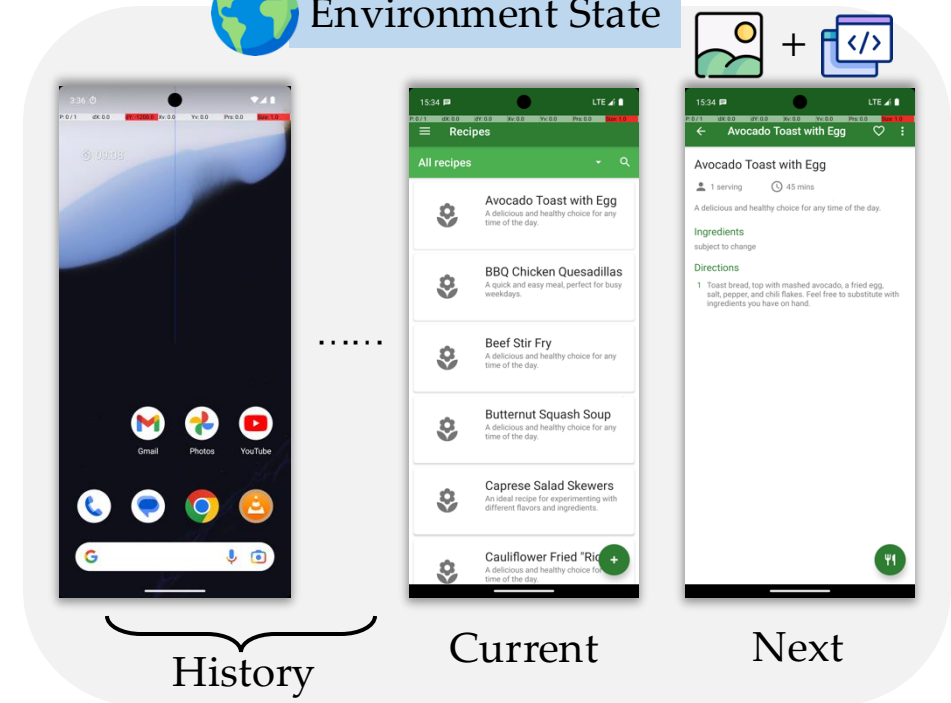
1. A **high-level instruction** that defines the overall goal the agent aims to accomplish
2. A series of **low-level instructions** that each describe specific steps required
3. **Actions** (e.g., CLICK, TYPE) 
4. **States**, which include visual representations like screenshots and textual representations such as a11ytree 

High-level Instruction

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



Environment State



Low-level Instruction

I need to click "Avocado Toast with Egg" to view more details and find the option to mark it as a favorite.

Action

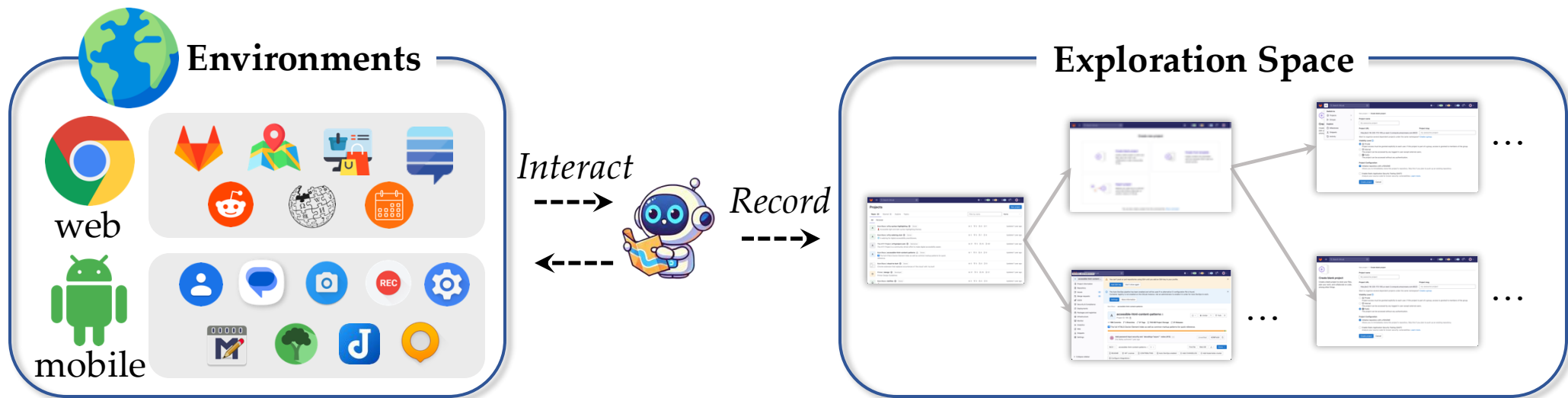
CLICK [Avocado Toast with Egg]
(698, 528)



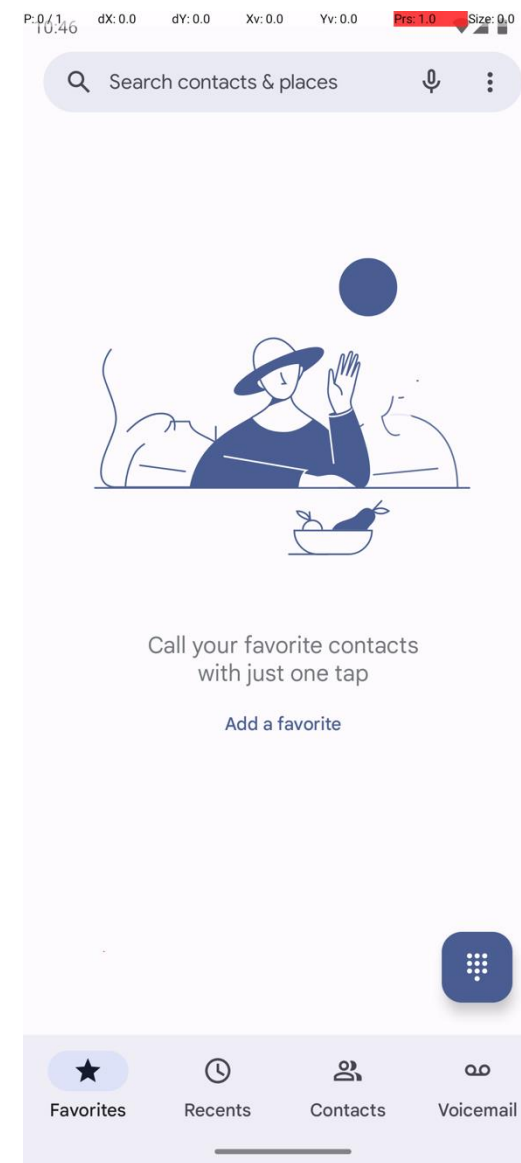
Reverse Task Synthesis

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: $\langle \text{Screen1}, \text{action}, \text{Screen2} \rangle$



Dynamic Environments



Dynamic Environments



My Account My Wish List Sign Out Welcome to One Stop Market

One Stop Market Search entire store here... [Advanced Search](#)

Beauty & Personal Care - Sports & Outdoors - Clothing, Shoes & Jewelry - Home & Kitchen - Office Products - Tools & Home Improvement -
Health & Household - Patio, Lawn & Garden - Electronics - **Cell Phones & Accessories** - Video Games - Grocery & Gourmet Food -

Home > Cell Phones & Accessories

Cell Phones & Accessories

Shop By Items 1-12 of 2449 Sort By

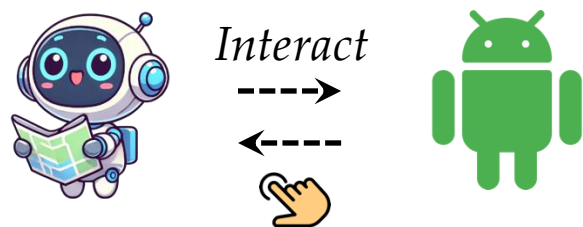
Shopping Options
Category
[Accessories\(1924\)](#)
[Cases, Holsters & Sleeves\(457\)](#)
[Cell Phones\(68\)](#)

Price
[\\$0.00 - \\$999.99\(2446\)](#)
[\\$1,000.00 and above\(3\)](#)

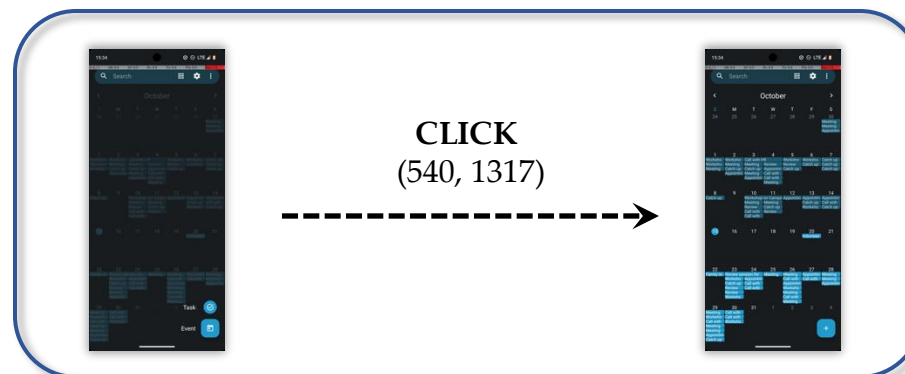
[Compare Products](#)

Reverse Task Synthesis

Retroactively interpreting changes in the GUI environment caused by actions.



Screenshots & Actions



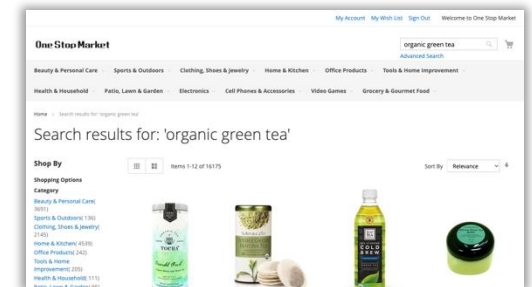
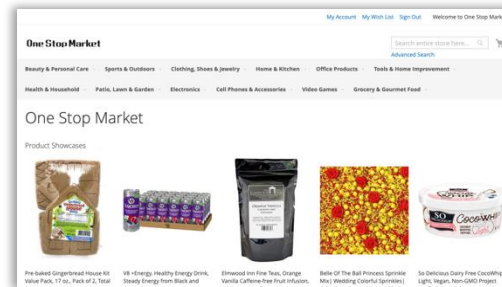
Reverse Task Synthesis

Retroactively interpreting changes in the GUI environment caused by actions.

Screenshots & Actions



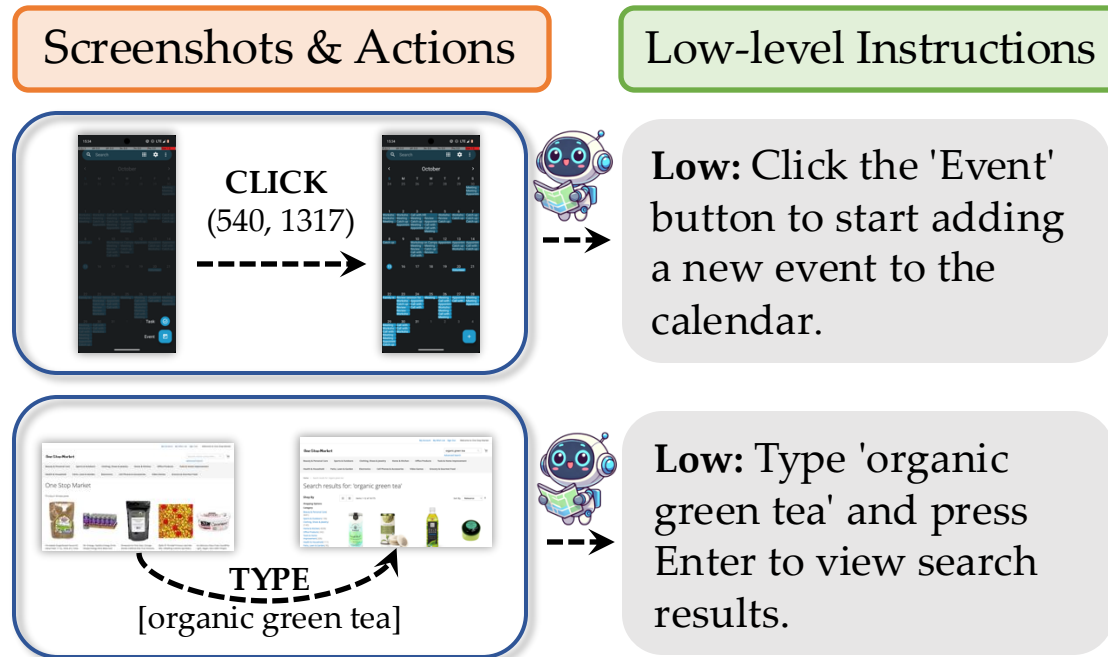
Interact



TYPE
[organic green tea]

Reverse Task Synthesis

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



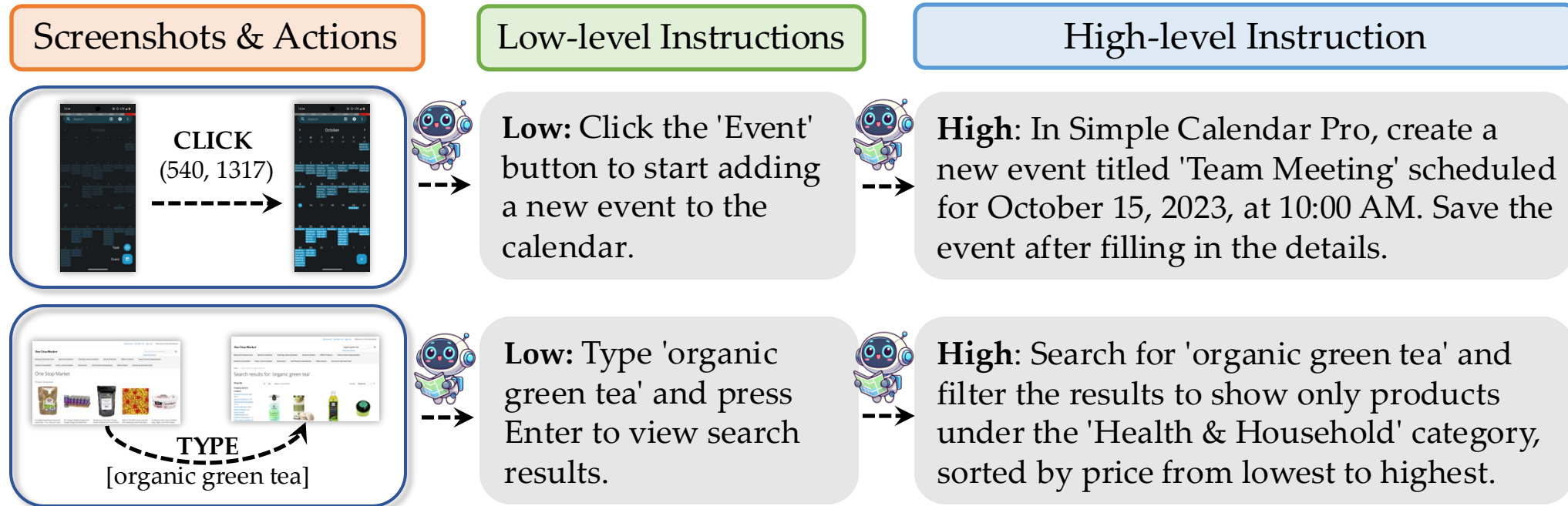
The data we synthesized:

1. Grounded

2. Actionable

Reverse Task Synthesis

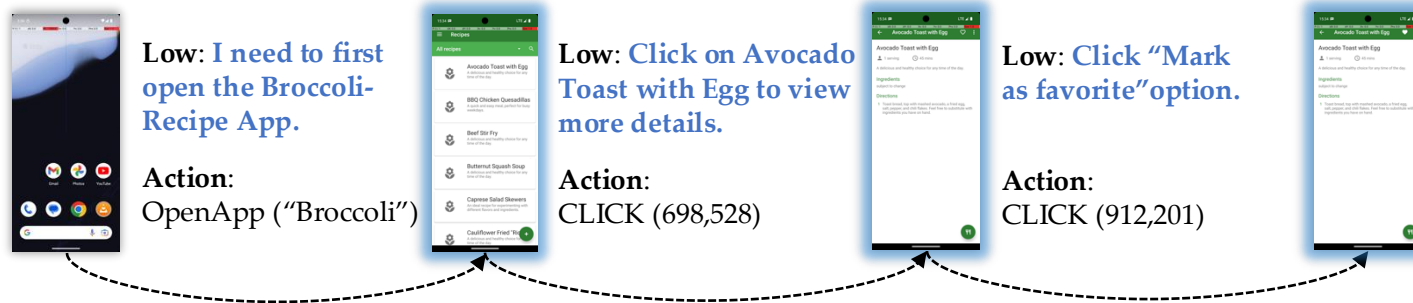
Retroactively interpreting changes in the GUI environment caused by actions, this process generates executable low-level instructions, which are then transformed into broader, goal-oriented high-level tasks



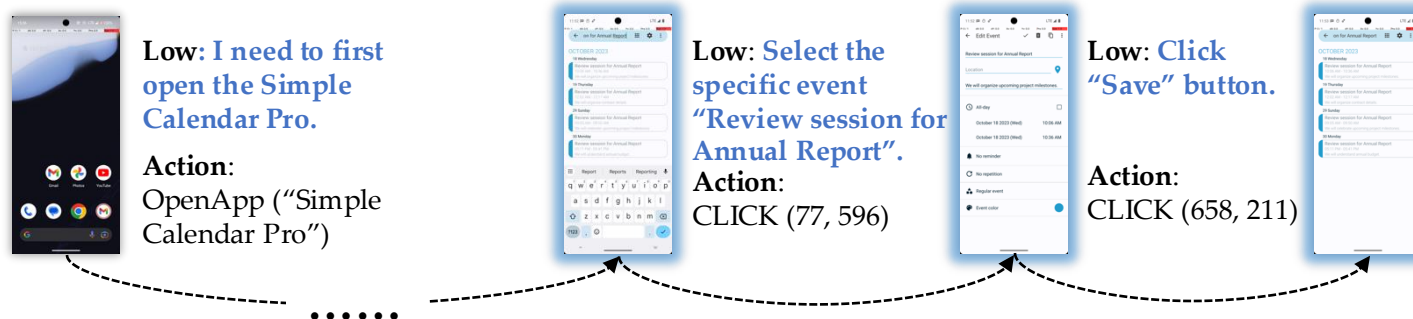
Reverse Task Synthesis

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.



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

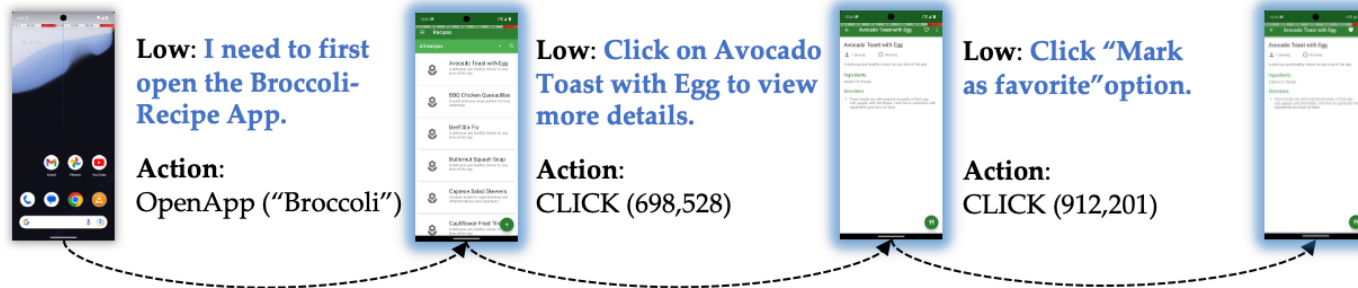


Reverse Task Synthesis

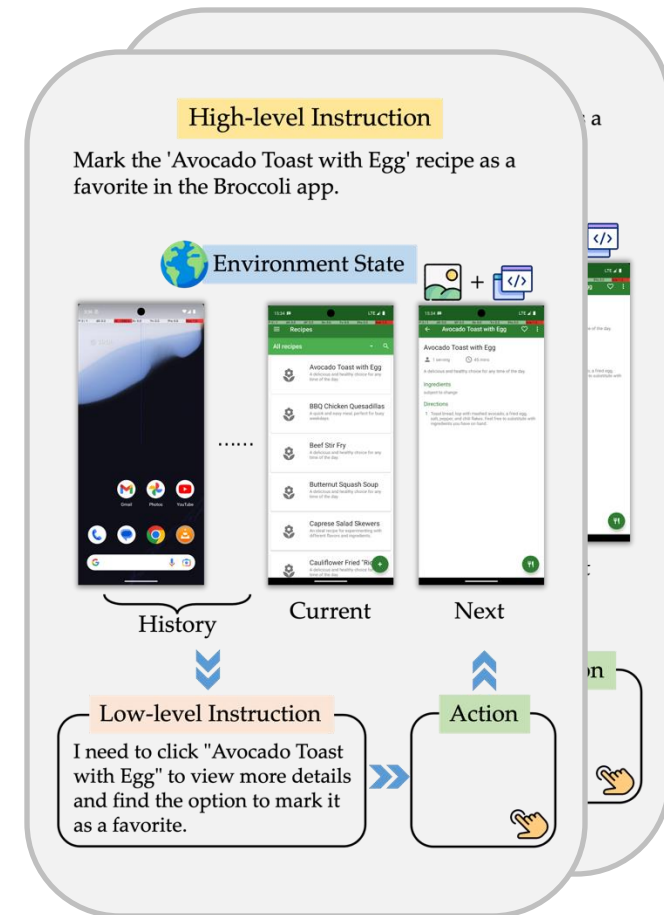
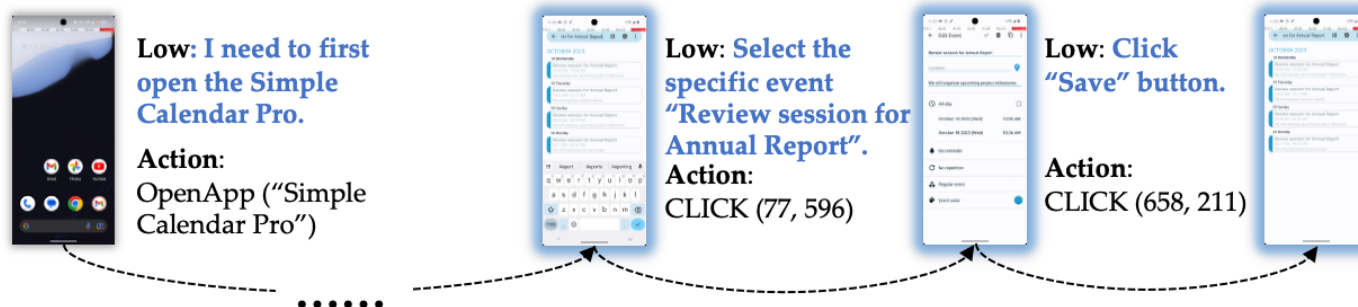
Trajectories collected! But is this all?

Let's consider data **quality** and synthesis **efficiency**.

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



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



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.

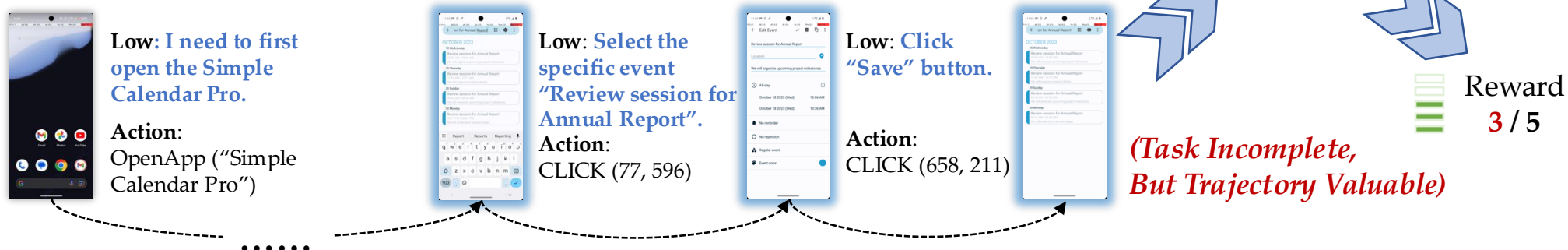
Reward Modeling

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

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



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



Models

Data Synthesis



GPT-4o



Qwen-VL Qwen2-VL-72B-Instruct

Backbones



InternVL InternVL2-4B / 8B

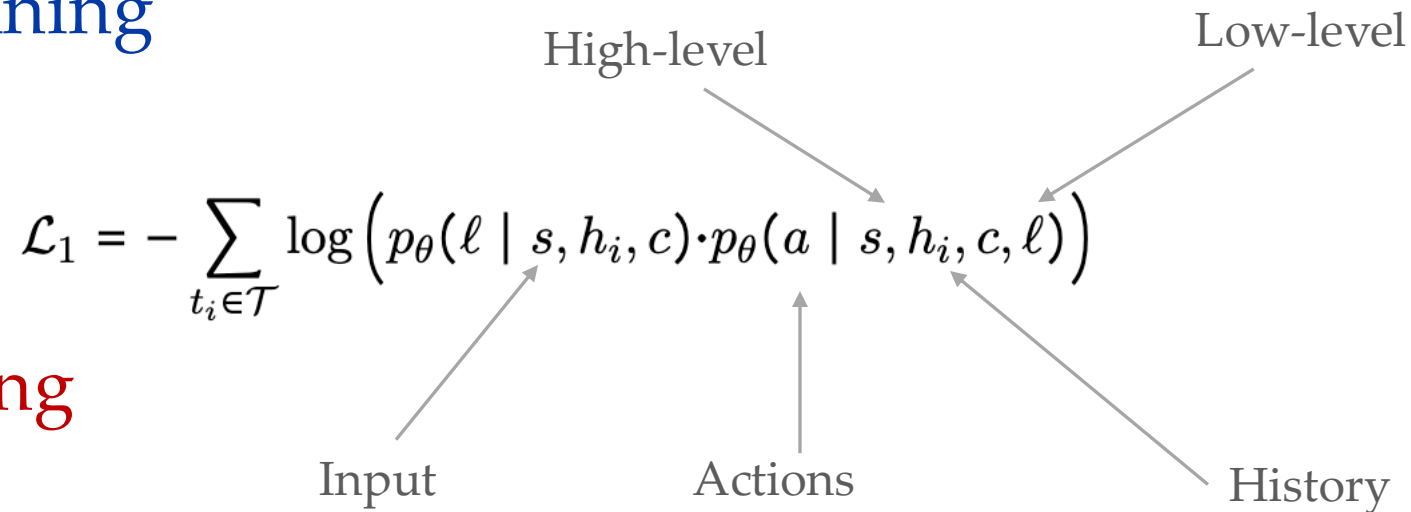


Qwen-VL Qwen2-VL-7B-Instruct

Training Strategies

Leverage **trajectory characteristics** to train GUI agents with complete capabilities

1. Planning Training



2. Action Training

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

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 }
```

...

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

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
	Zero-Shot	2.23	17.89	38.22	47.69	66.67
InternVL2-8B	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
	Zero-Shot	0.89	28.92	61.39	46.37	72.78
Qwen2-VL-7B	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

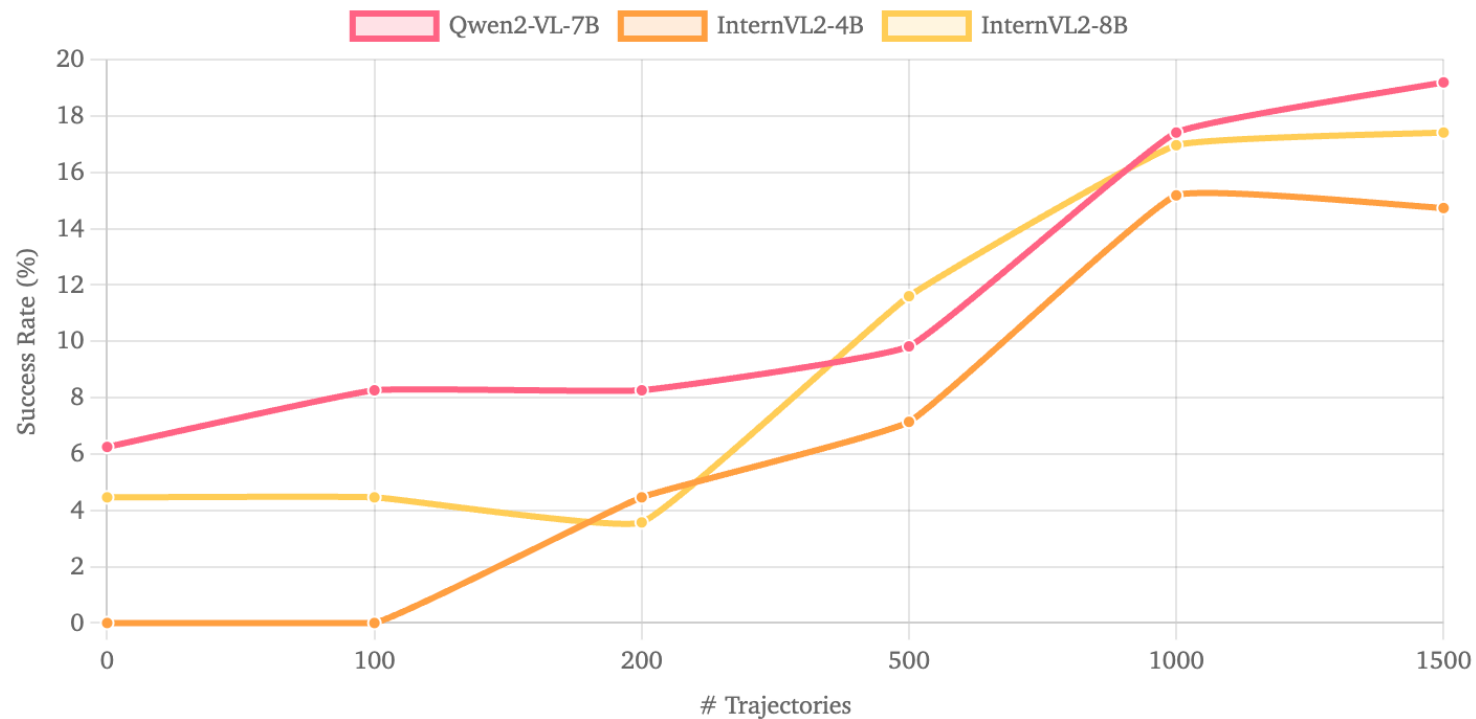
Experiments: Web

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
	Zero-Shot	0.00	0.00	0.00	0.00	0.00	0.00
InternVL2-4B	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 **Scaling** Trajectory Data Improves Agentic Ability?

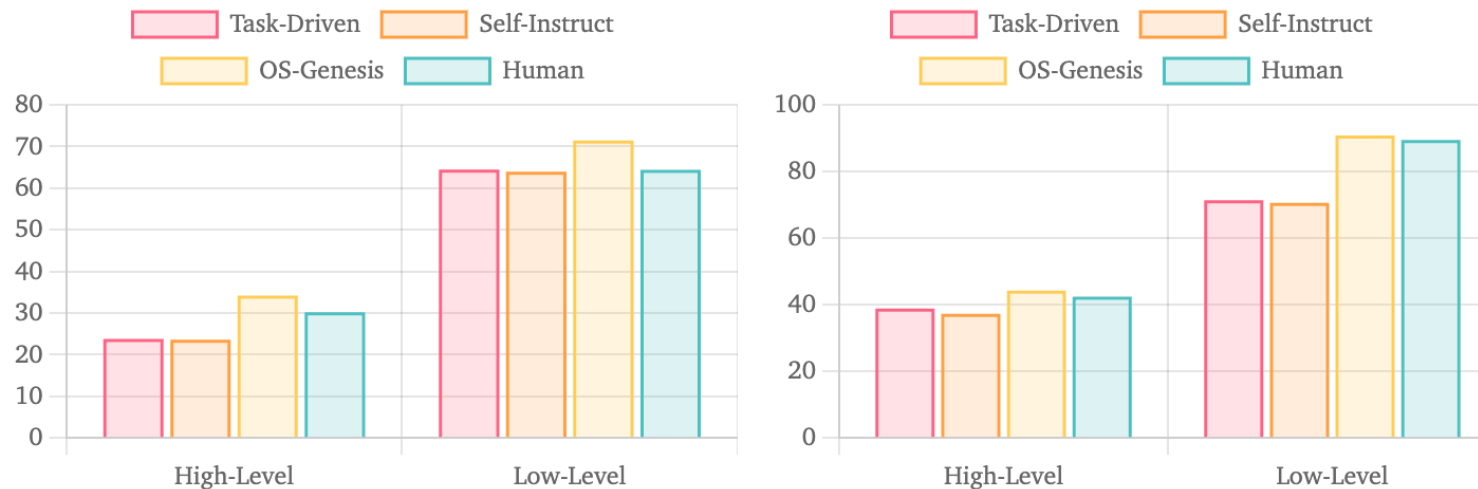


Insight: Generally improves, but will **saturate**.

Analysis

How Far are we from **Human Data**?

Let's first take a look at **high-level instructions**.



Insight: Reverse Task Synthesis Elicits Better **Executability**.

Analysis

How Far are we from **Human Data**?

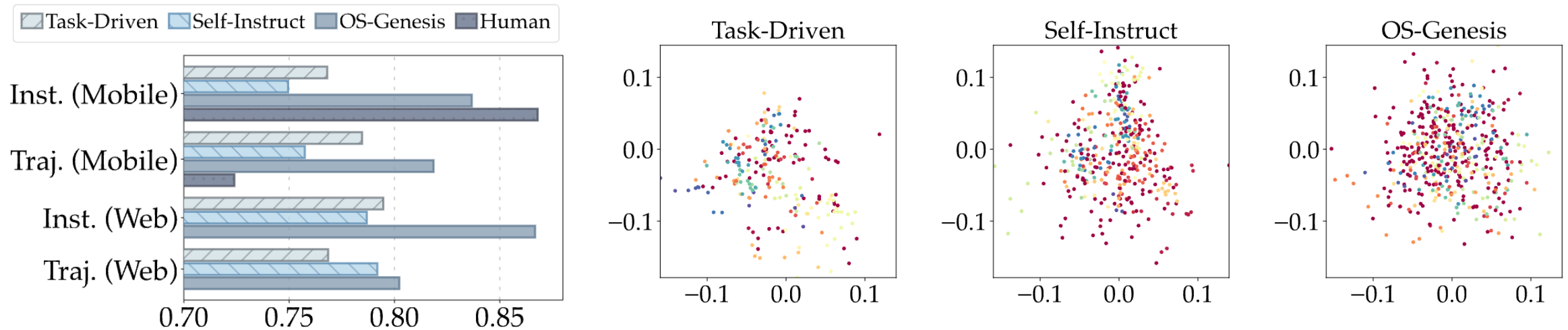
Then, OS-Genesis v.s. **Human-annotated Trajectories**.



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 Hugging Face

The screenshot shows a Hugging Face post for a paper. At the top, the Hugging Face logo and navigation menu are visible. The post title is "OS-Genesis: Automating GUI Agent Trajectory Construction via Reverse Task Synthesis". It is published on Dec 28, 2024, and submitted by Qiushi Sun on Jan 2, 2025, with a "#1 Paper of the day" badge. The authors listed are Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhouchuanze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, and Zhiyong Wu. The abstract begins with "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..." On the right side, there is a section for "Models citing this paper" with 9 models listed: OS-Copilot/OS-Genesis-4B-AC, OS-Copilot/OS-Genesis-7B-AC, OS-Copilot/OS-Genesis-8B-AC, and OS-Copilot/OS-Genesis-4B-AW. Each model entry includes its name, the task "Image-Text-to-Text", the update date, and the number of downloads and likes.

Hugging Face Search models, datasets, users... Models Datasets Spaces Posts Docs Enterprise Pricing

< Papers arxiv:2412.19723

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

Published on Dec 28, 2024 · Submitted by [Qiushi Sun](#) 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](#), Zhouchuanze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, Zhiyong Wu

Abstract

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...

▲ Upvoted 82


Models citing this paper 9

- OS-Copilot/OS-Genesis-4B-AC
Image-Text-to-Text · Updated Jan 8 · 50 · 7
- OS-Copilot/OS-Genesis-7B-AC
Image-Text-to-Text · Updated Jan 8 · 69 · 6
- OS-Copilot/OS-Genesis-8B-AC
Image-Text-to-Text · Updated Jan 8 · 48 · 4
- OS-Copilot/OS-Genesis-4B-AW
Image-Text-to-Text · Updated Jan 6 · 31

[Browse 9 models citing this paper](#)

Checkpoints & Data Access

Available on  ModelScope

 **OS-Copilot** 研
取消关注 通知设置 申请审批中...

全部 17 合集 0 模型 13 数据集 4 创空间 0 品牌馆 0

模型

最近更新 11

OS-Genesis-7B-WA
OS-Copilot/OS-Genesis-7B-WA
视觉多模态理解 Transformers, Safetensors等3个框架 qwen2_vl 开源协议: apache-2.0
OS-Copilot 2025.01.09 更新 | 241 | 0

OS-Genesis-7B-AW
OS-Copilot/OS-Genesis-7B-AW
视觉多模态理解 Transformers, Safetensors等3个框架 qwen2_vl 开源协议: apache-2.0
OS-Copilot 2025.01.09 更新 | 235 | 0

OS-Genesis-8B-AC
OS-Copilot/OS-Genesis-8B-AC
视觉多模态理解 PyTorch, Transformers等3个框架 internvl_chat 开源协议: apache-2.0
OS-Copilot 2025.01.09 更新 | 285 | 0

OS-Genesis-4B-AC
OS-Copilot/OS-Genesis-4B-AC
视觉多模态理解 PyTorch, Transformers等3个框架 internvl_chat 开源协议: apache-2.0
OS-Copilot 2025.01.09 更新 | 279 | 0

OS-Genesis-7B-AC
OS-Copilot/OS-Genesis-7B-AC
视觉多模态理解 Transformers, Safetensors等3个框架 qwen2_vl 开源协议: apache-2.0
OS-Copilot 2025.01.08 更新 | 287 | 1

OS-Genesis-8B-WA
OS-Copilot/OS-Genesis-8B-WA
视觉多模态理解 PyTorch, Transformers等3个框架 internvl_chat 开源协议: apache-2.0
OS-Copilot 2025.01.08 更新 | 239 | 0

OS-Genesis-4B-AW
OS-Copilot/OS-Genesis-4B-AW
视觉多模态理解 PyTorch, Transformers等3个框架 internvl_chat 开源协议: apache-2.0
OS-Copilot 2025.01.08 更新 | 242 | 0

OS-Genesis-8B-AW
OS-Copilot/OS-Genesis-8B-AW
视觉多模态理解 PyTorch, Transformers等3个框架 internvl_chat 开源协议: apache-2.0
OS-Copilot 2025.01.08 更新 | 245 | 0

关于我们

组织成员





您可以创建自己的组织 [申请创建](#)


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





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



SCHOOL OF
COMPUTING &
DATA SCIENCE
The University of Hong Kong

Thanks for listening

Contact: qiushisun@connect.hku.hk