COMP7607 Final Project:
Foundation Agents for General Digital Control

Zhiyong Wu

2023/09/09

1. Final Project Announcement

2. Background & Research Sharing

3. Exemplary Projects

Self-Introduction: Zhiyong Wu

Education

- 2017-2021，PhD，University of Hong Kong
- 2013-2017，BEng, Wuhan University

Experience

- 2021.12- now，Shanghai AI Lab，Research Scientist

Research Interest

- Large Language Models
- Language Agent

Jarvis From the Iron Man

COMP7607 Final Project: Building Your Own JARVIS

Teams:

5-8 students

Topics:

Create a fully functional OS agent (JARVIS-like digital copilot)

COMP7607: Building Jarvis-like Digital Copilot

Emboided AI,

but in the virtual world!

Value:

Improve the productivity of the whole sociaty

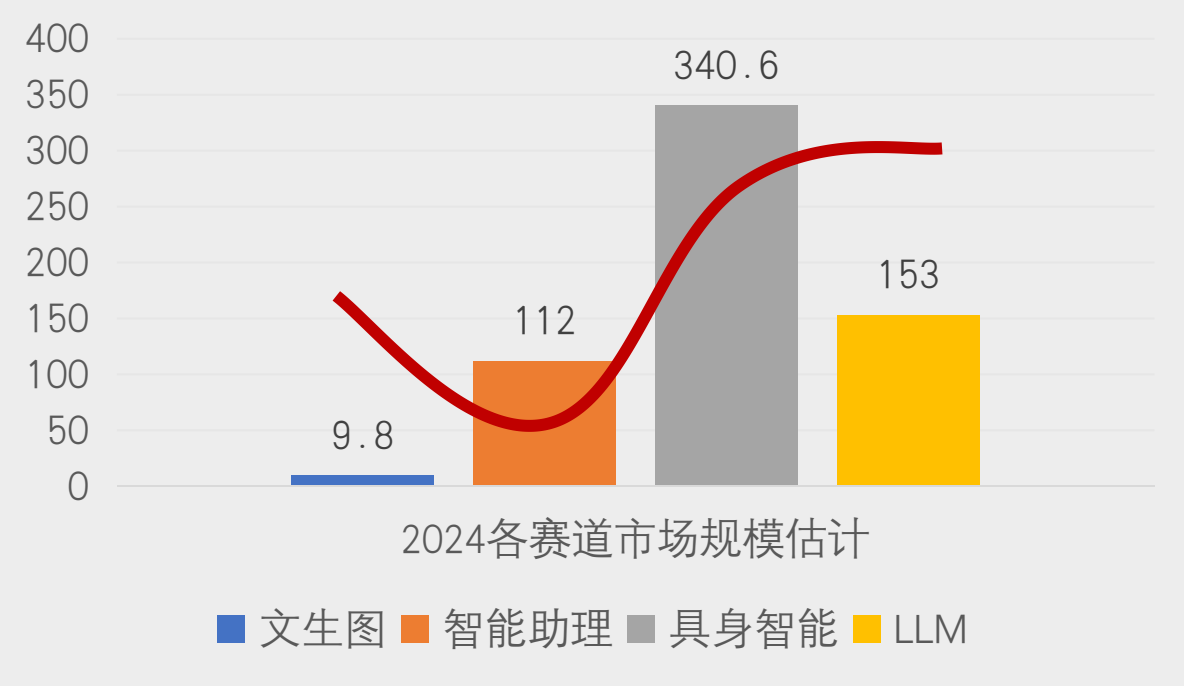
Enhance technology accessibility for the elderly or individuals with lower education levels.

WHY

WHY: An 'underestimated' track where the research attention **(red curve)** is not proportional to the capital investment.

=》 The opportunity to become a field leader ↑

=> Lots of job opp in industry



Text-to-Image: Stable Diffusion, Dall-E

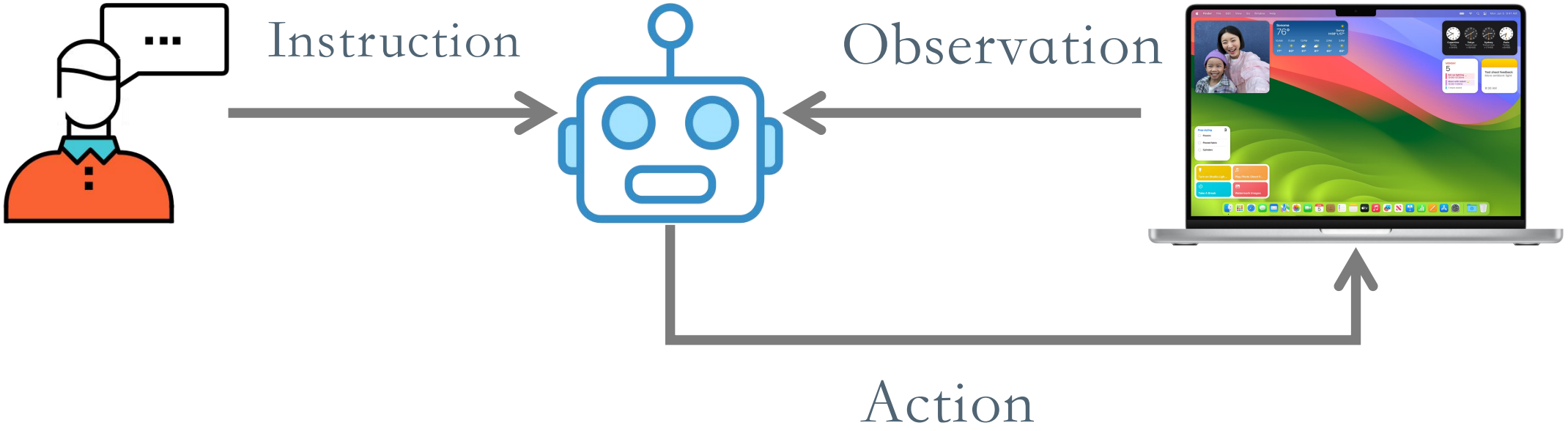
Embodied AI: Stanford、Berkeley

LLM: OpenAI, Anthotopic

Digital Assistant: ?

Siri, Cortana, Alexa

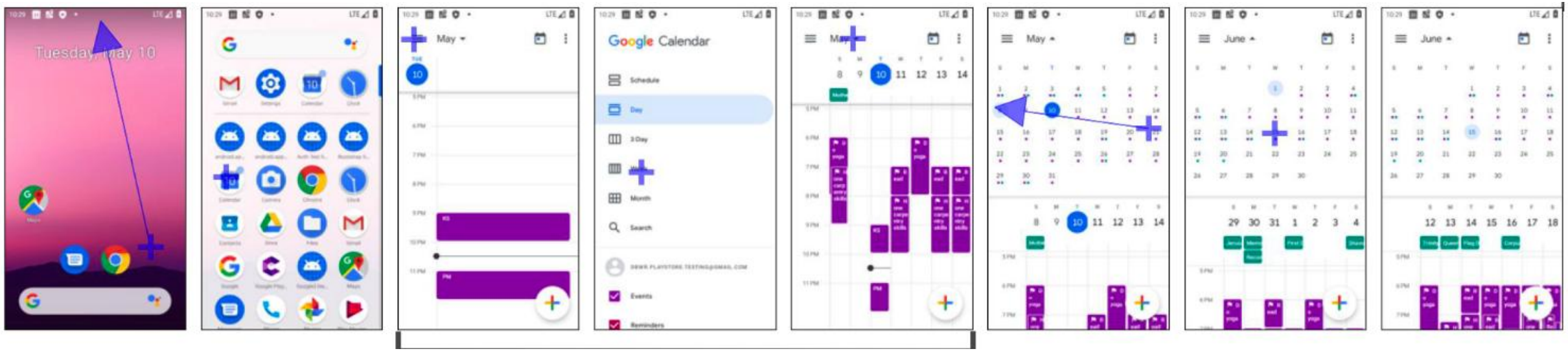
OS Agent System Overview



Challenge 1: Data Scarcity

Trajectory annotations are scarce and expensive

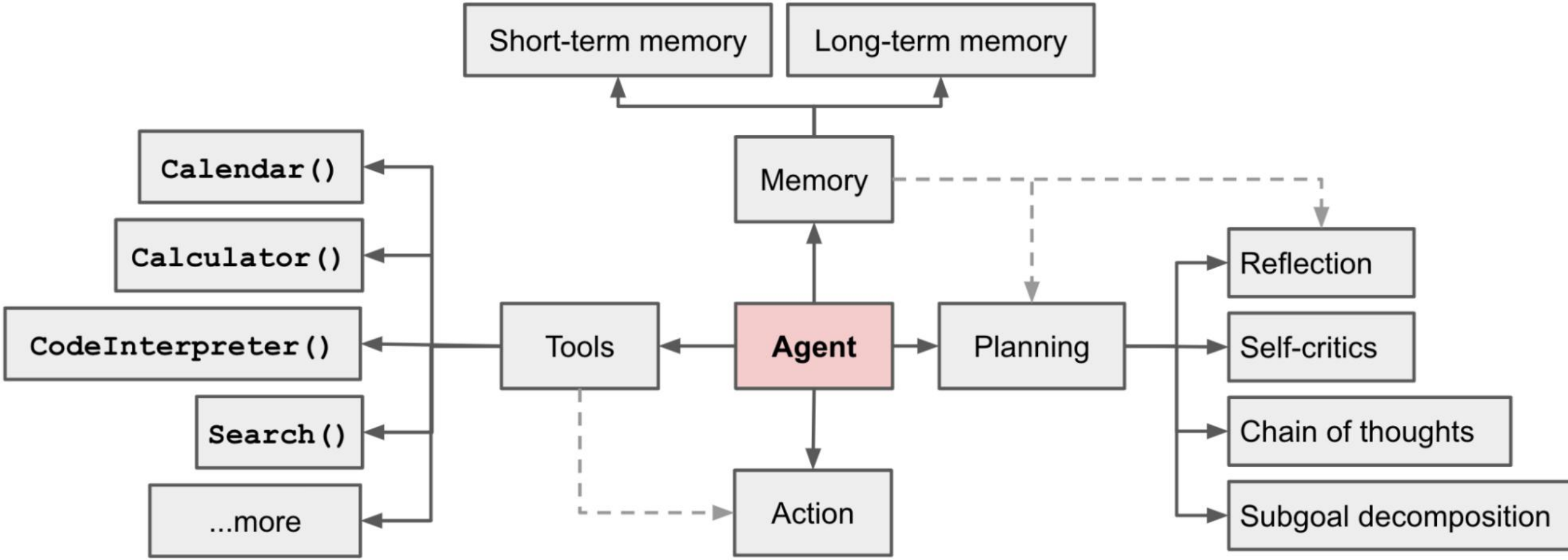
e.g., DeepMind/Android/15k traj/833 app/20 ppl/4 mo



On the Effects of Data Scale on Computer Control Agents. Li et al., 2024

Challenge 2: from foundation model to ACTION model

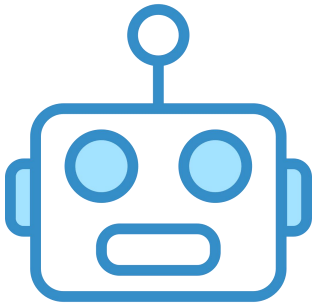
Foundation models can read and write, but they can't act



LLM Powered Autonomous Agents. Lilien Weng, 2023

Challenge 3: Safety and Interpretabilitiy

Help me clean my desktop



```
cd Desktop
rm -rf ./
```

Web agent for shopping

Research Challenge

1. Scarcity of data
   Trajectory annotations are scarce and expensive
   e.g., DeepMind/Android/15k traj/833 app/20 ppl/4 mo

2. Foundation models can read and write, but they can't act
   Special traning is needed for digital grounding

3. Safety and Interpretability
   Lack of transprancy require more user trust

Research Challenge

1. Scarcity of data
   Trajectory annotations are scarce and expensive
   e.g., DeepMind/Android/15k traj/833 app/20 ppl/4 mo
2. Foundation models can read and write, but can't act
   Specific traning is needed for digital grounding
   => Executable Language Grounding
3. Safety and Interpretability
   Lack of transprancy require more user trust

Executable Language Grounding

Symbol-LLM (ACL'24) OS-Copilot (ICLR'24 WS)

1. Grounding via code： turn natural language instructions into executable programs

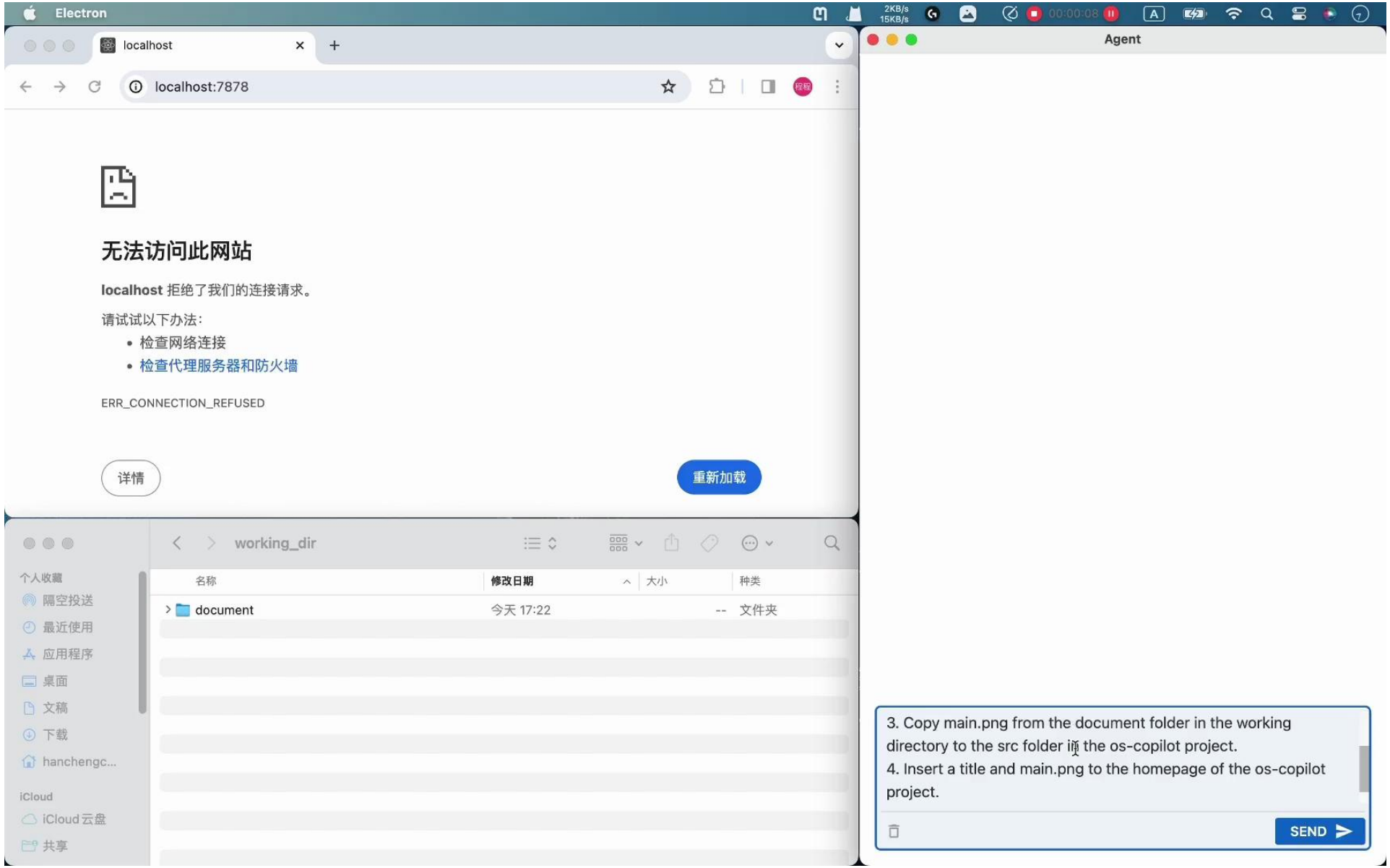
Example: "Please switch my computer to dark mode."

applescript 复制

```
tell application "System Events"
    tell appearance preferences
        set appearance to dark
    end tell
end tell
```

OS-Copilot: Towards Generalist Computer Agents with Self-Improvement. Wu et al., 2024

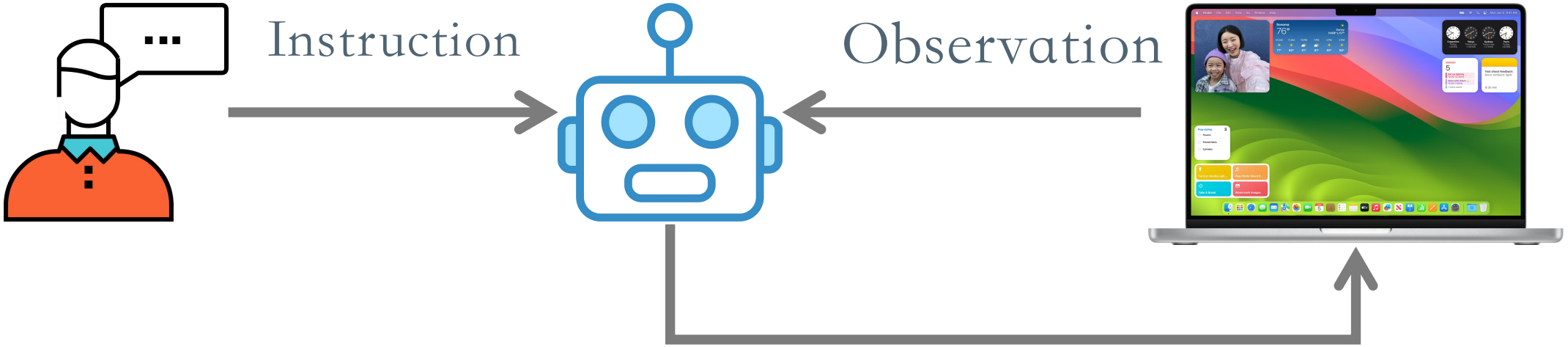
Demo: Building a React Website



OS-Copilot System Overview

"Please switch my computer to dark mode."

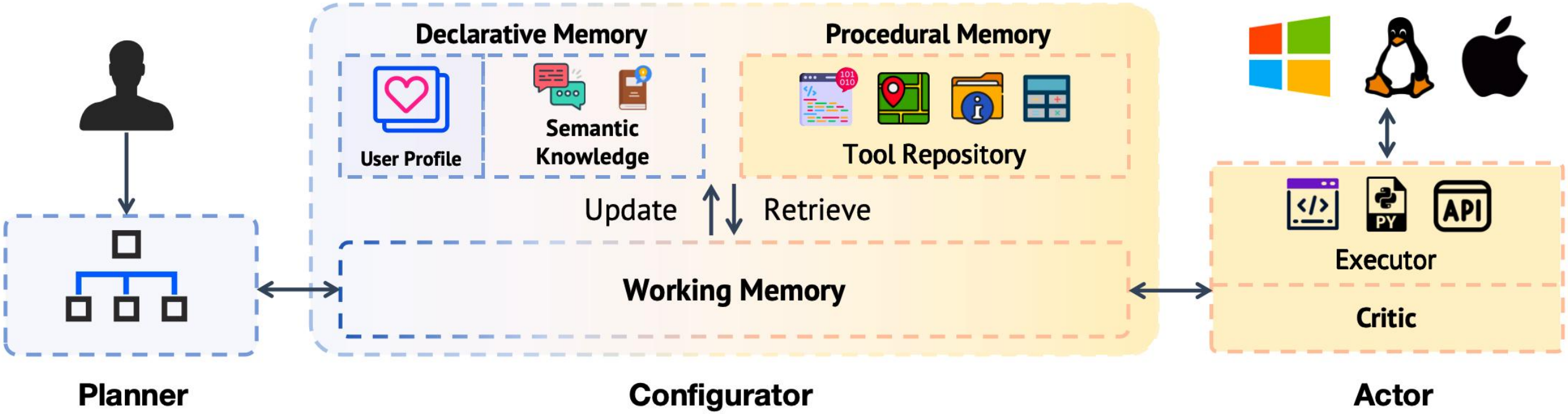
pwd # get current directory
ls # see what's in the folder
capture cmd output



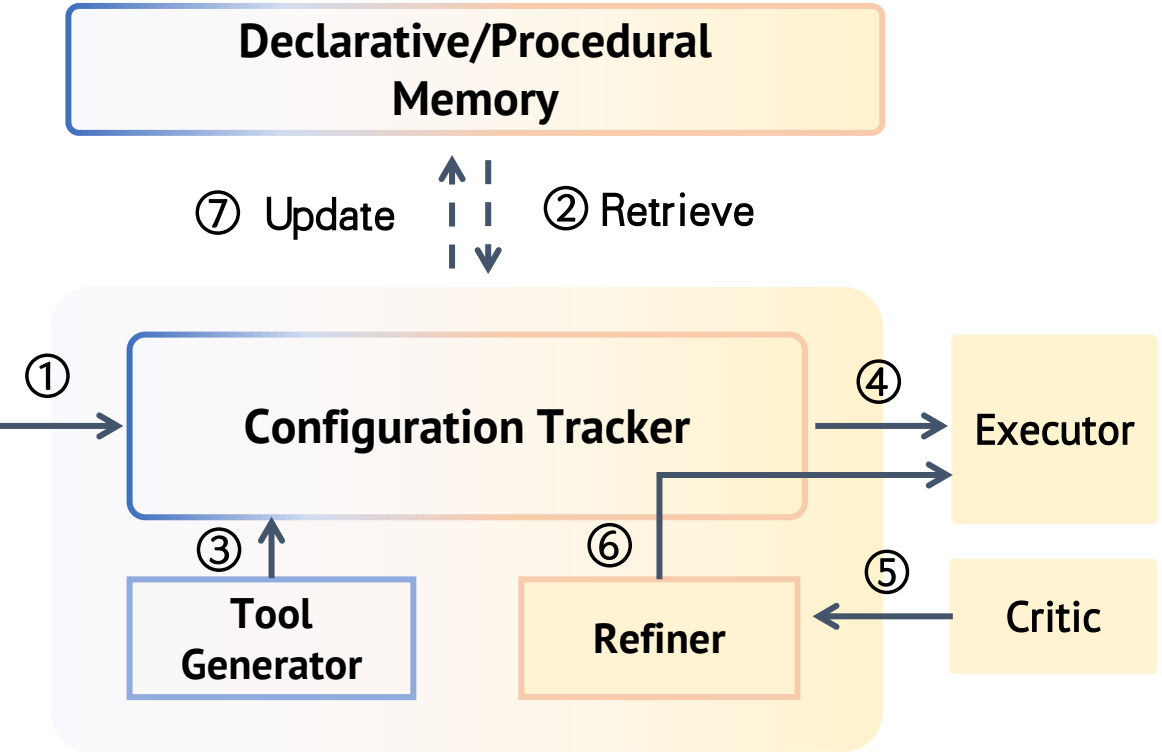
applescript 复制

```applescript
tell application "System Events"
    tell appearance preferences
        set appearance to dark
    end tell
end tell
```

OS-Copilot System Overview



OS-Copilot: Configurator



Subtask: Change the system into the Dark mode

Tool Generator:

```
class change_system_appearance(BaseAction):
        ....
        script = 'tell app "System Events" to tell appearance preferences to set dark mode to true'
        ...
```

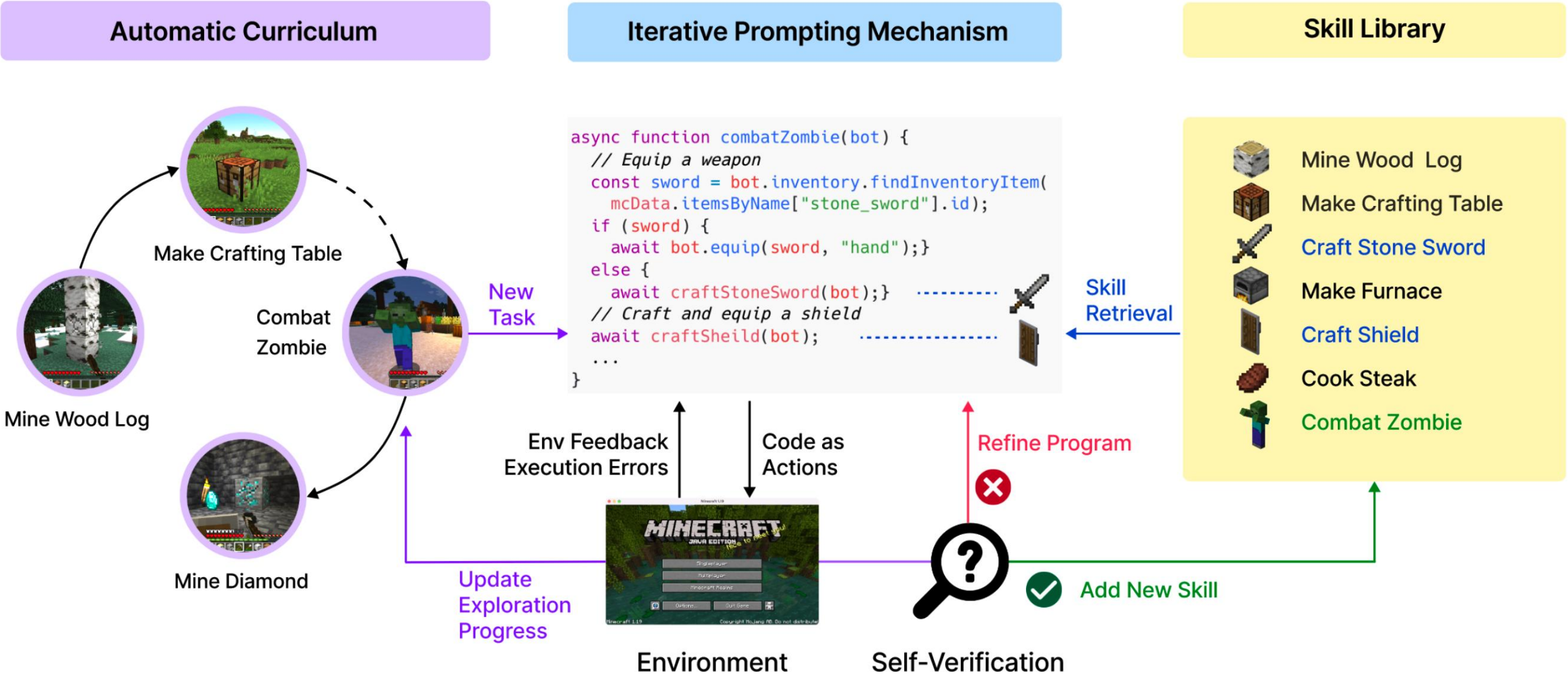
Executor:

(i) Save the tool to change_system_appearance.py

(ii) Execute the tool

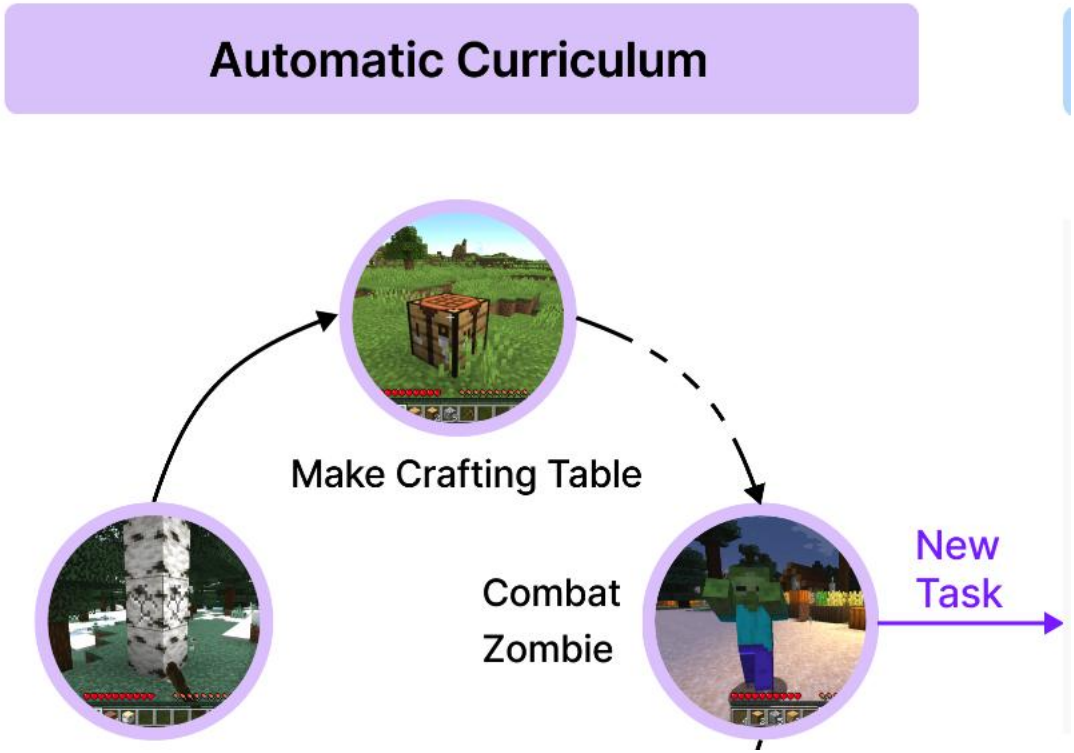
```
>_ python change_system_appearance.py dark
```

Populate the memory through self-directed learning



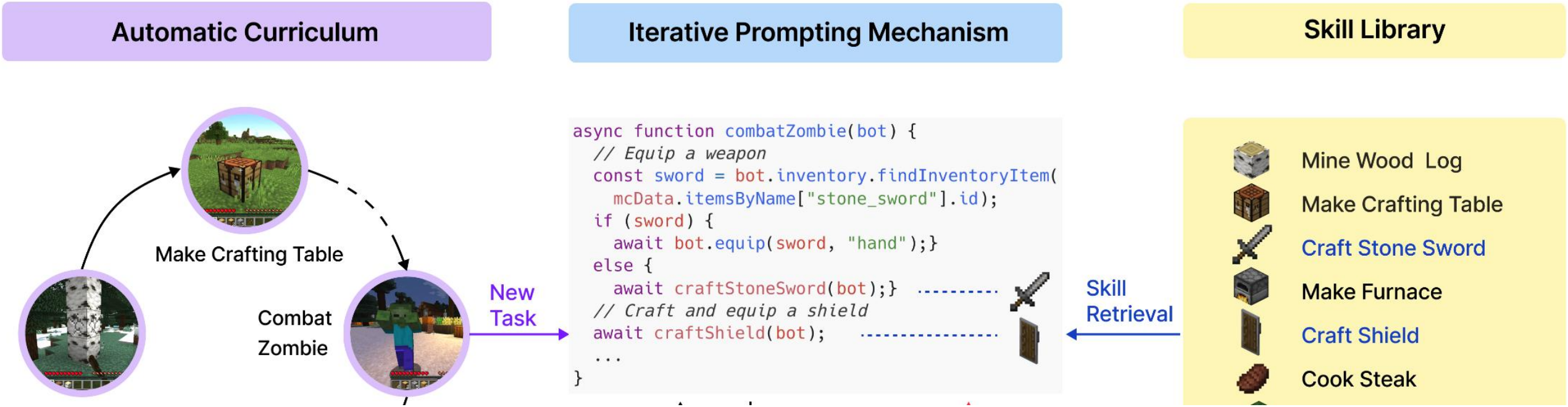
VOYAGER: An Open-Ended Embodied Agent with Large Language Models. Wang et al., 2023

Populate the memory through self-directed learning



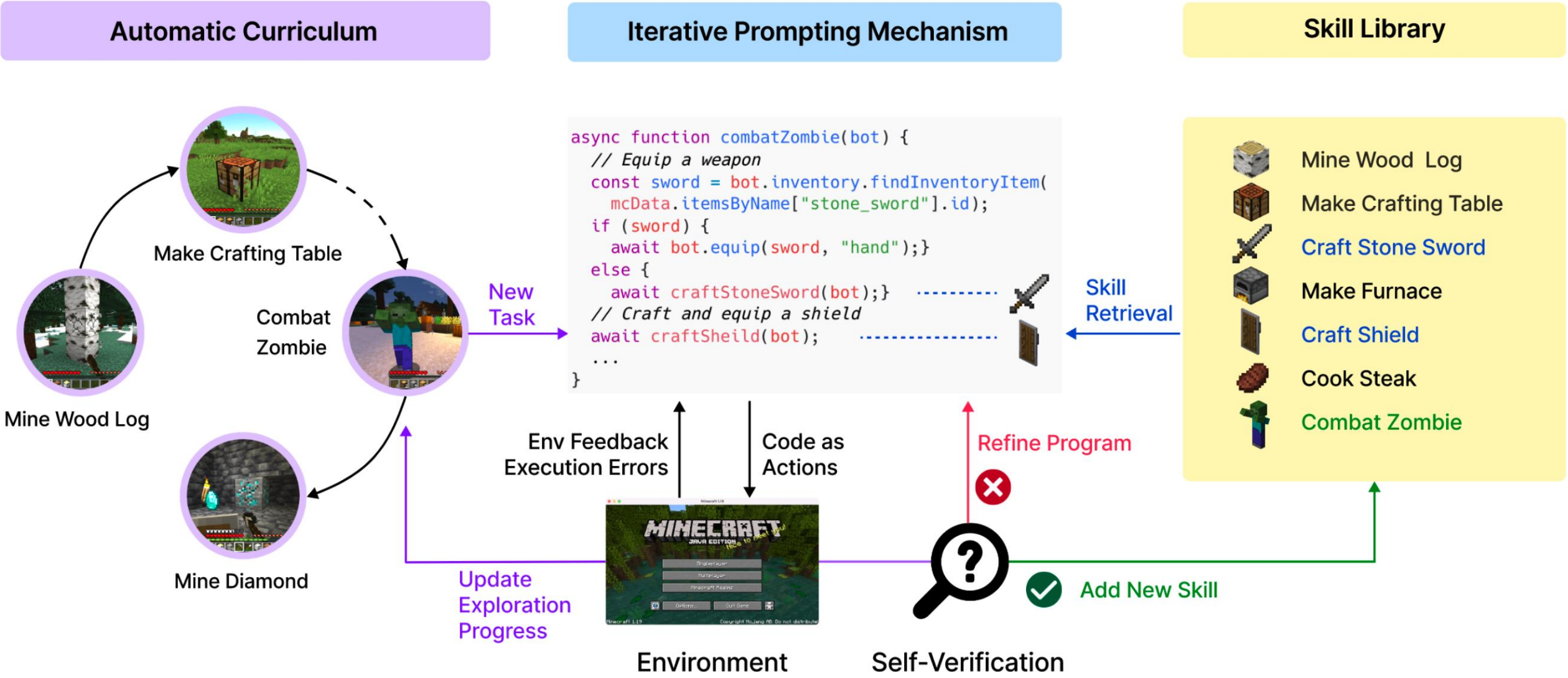
VOYAGER: An Open-Ended Embodied Agent with Large Language Models. Wang et al., 2023

Populate the memory through self-directed learning



VOYAGER: An Open-Ended Embodied Agent with Large Language Models. Wang et al., 2023

Populate the memory through self-directed learning



VOYAGER: An Open-Ended Embodied Agent with Large Language Models. Wang et al., 2023

1. Grounding via Code

Symbol-LLM (ACL'24) OS-Copilot (ICLR'24 WS)

1. Grounding via code：

Example: "Please switch my computer to dark mode."

applescript 复制

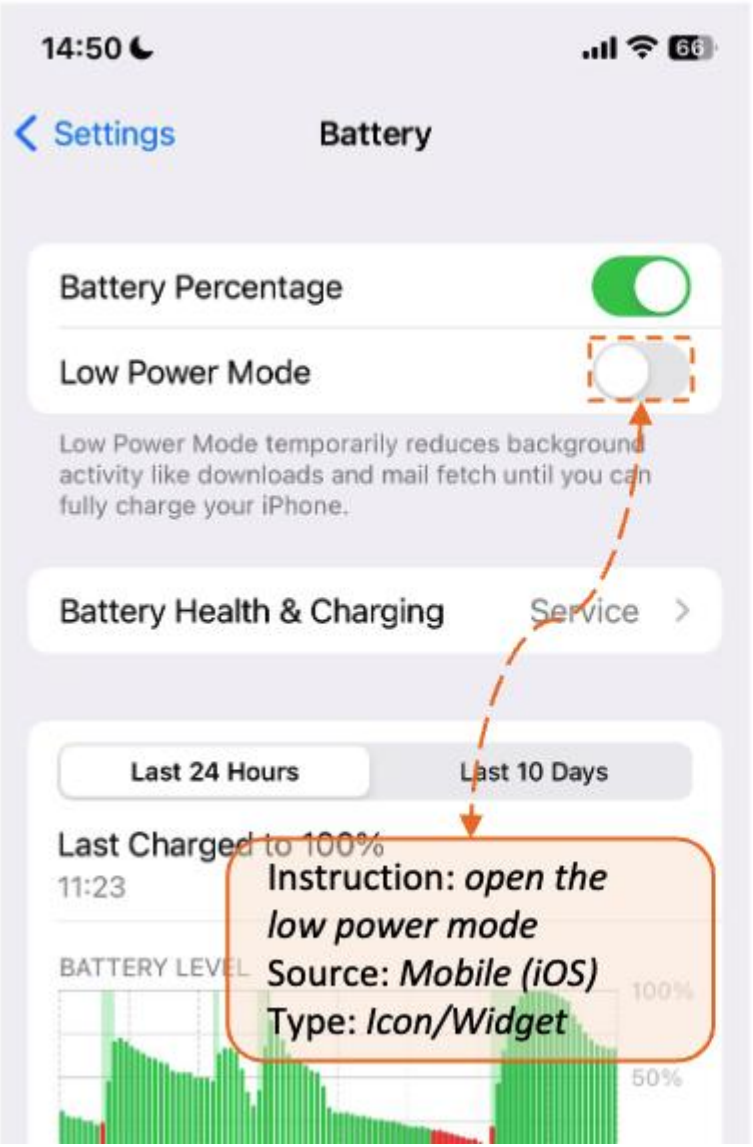
```
tell application "System Events"
    tell appearance preferences
        set appearance to dark
    end tell
end tell
```

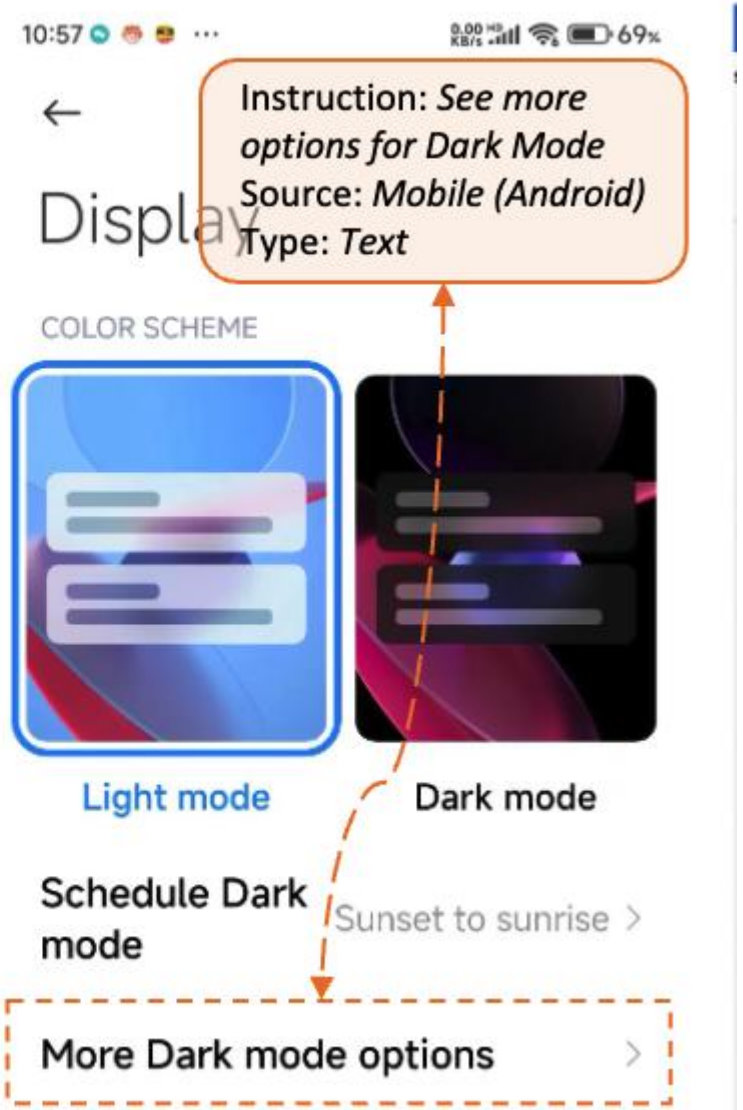
CON： Most software is not open sourced!

2. Grounding via GUI Interaction

2. Grounding via GUI interaction: Turn intstruction in human-like GUI interaction (e.g., Click)

Challenge: understand and locate GUI element

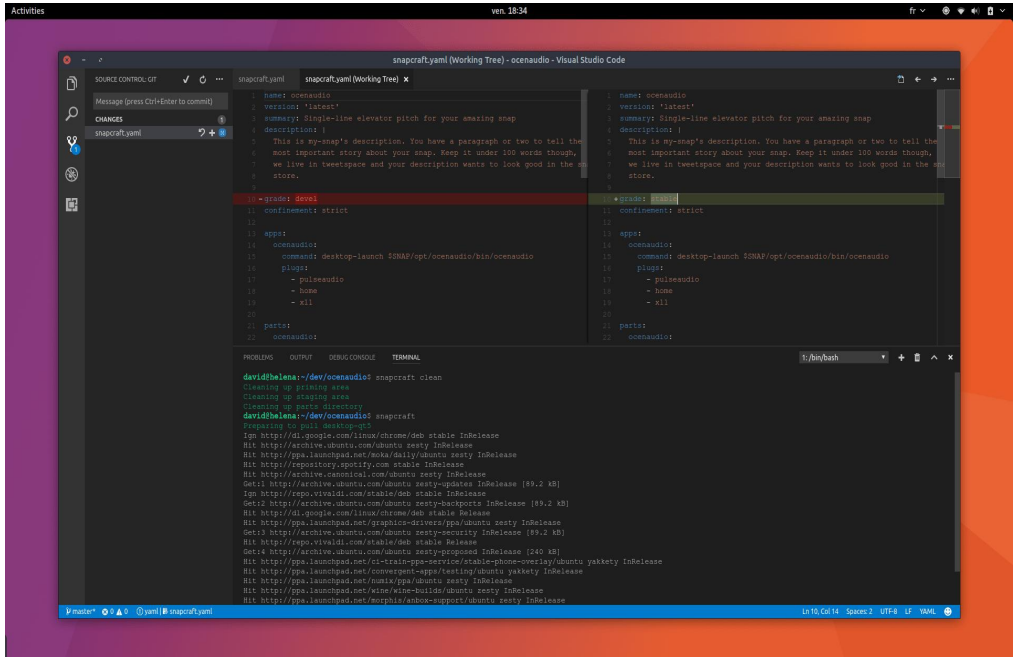




SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents. Cheng et al., 2024

2. Grounding via GUI Interaction

VS

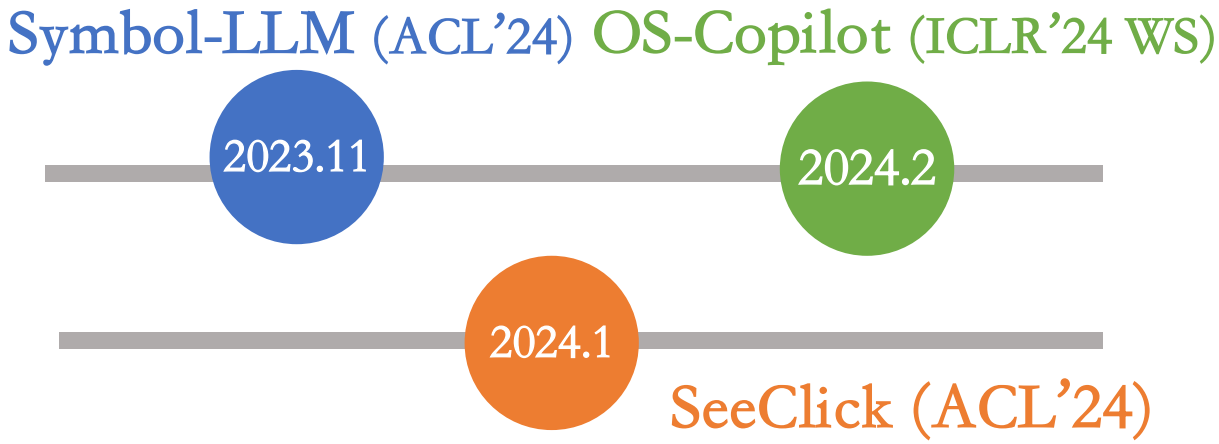


Current VLMs are mostly trained with natural images

OS Agent often deals with “unnatural” images

SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents. Cheng et al., 2024

Executable Language Grounding



CON: low-efficiency,
do we really need human-like agents?

	Pros	Cons
GUI	Highly generalizable	Low-efficiency, Upperbounded by current VLM
CLI	High-efficiency	Poor generalizability

COMP7607 Final Project: Building Your Own JARVIS

Teams:

5-8 students

Topics:

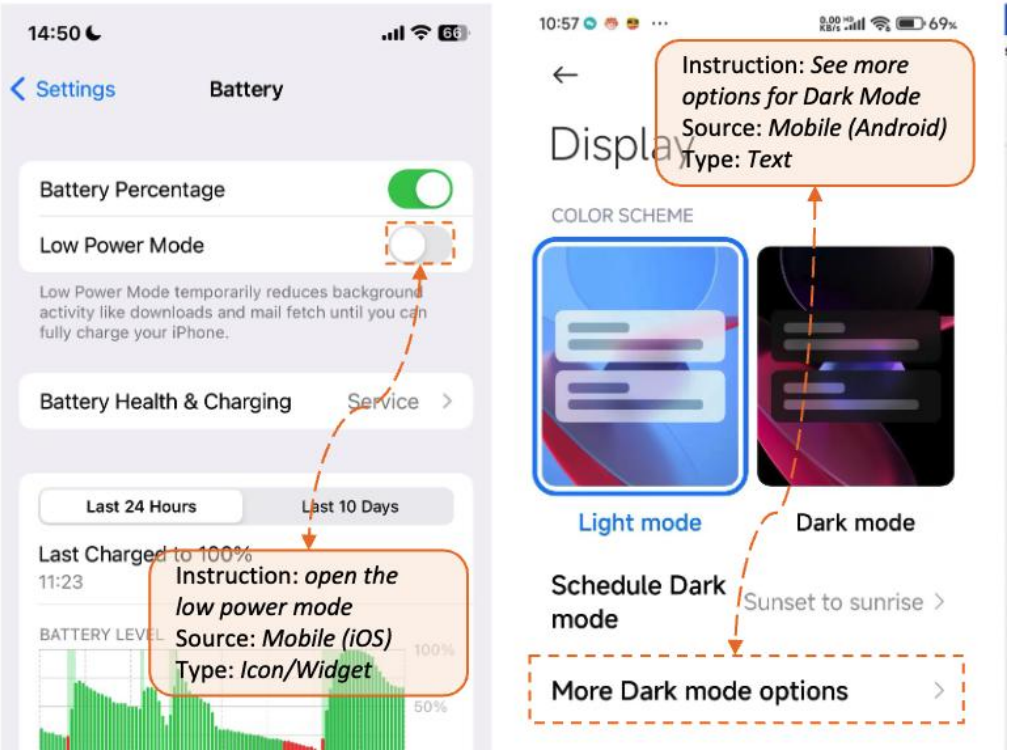
Using OS-Copilot as the base, creating a fully functional OS agent (JARVIS-like digital copilot)

Contributing to OS-Copilot by making a PR/MR!

| Exemplary Projects

Exemplary Project 1: Mobile Agent

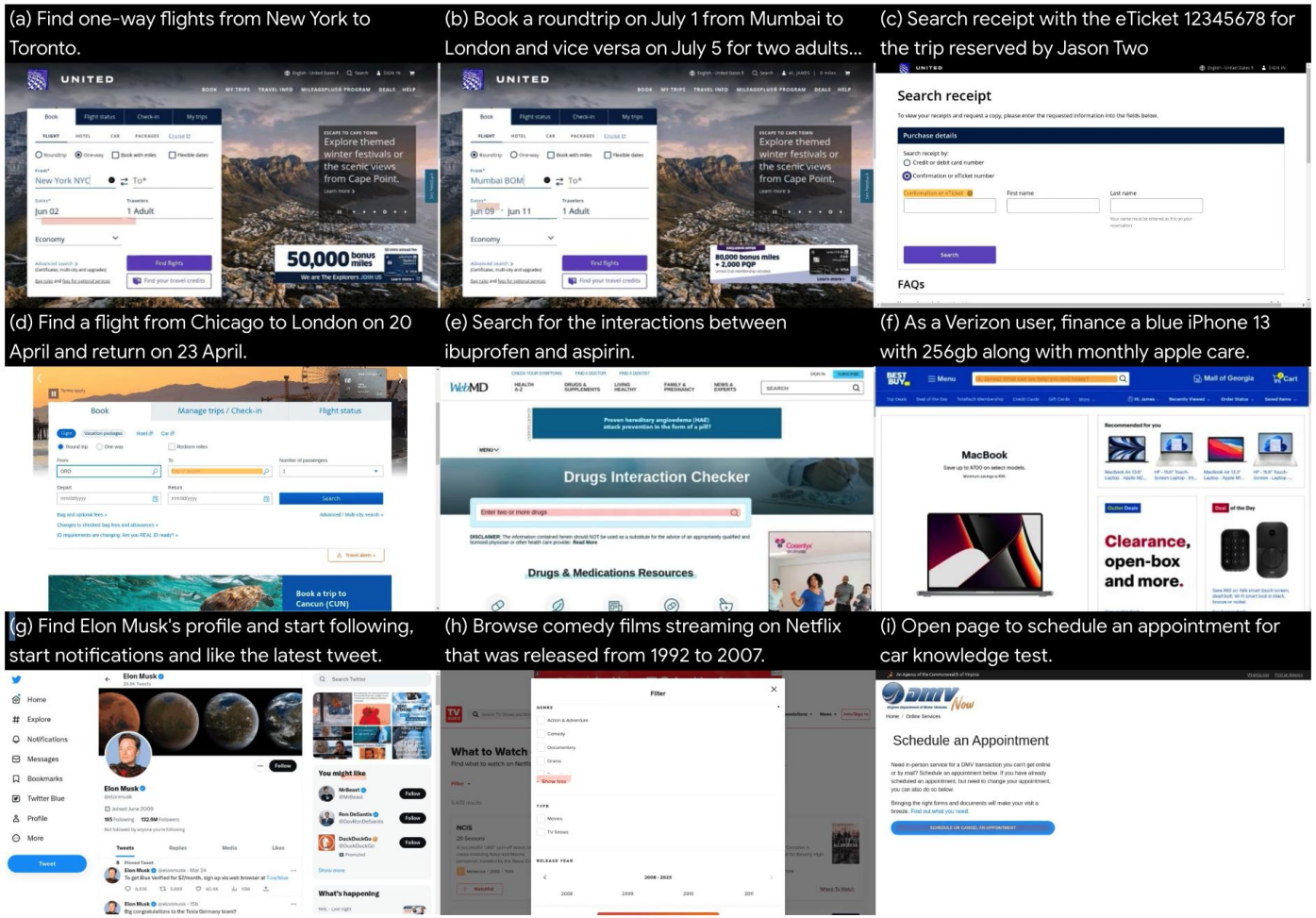
Extending the current OS-Copilot framework to support mobile control



1. SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents. Cheng et al., 2024
2. Mobile-Agent: The Powerful Mobile Device Operation Assistant Family. Wang et al., 2024
3. On the Effects of Data Scale on Computer Control Agents. Li et al., 2024
4. https://www.youtube.com/watch?v=EMbIpzqJld0

Exemplary Project 2: Web Agent

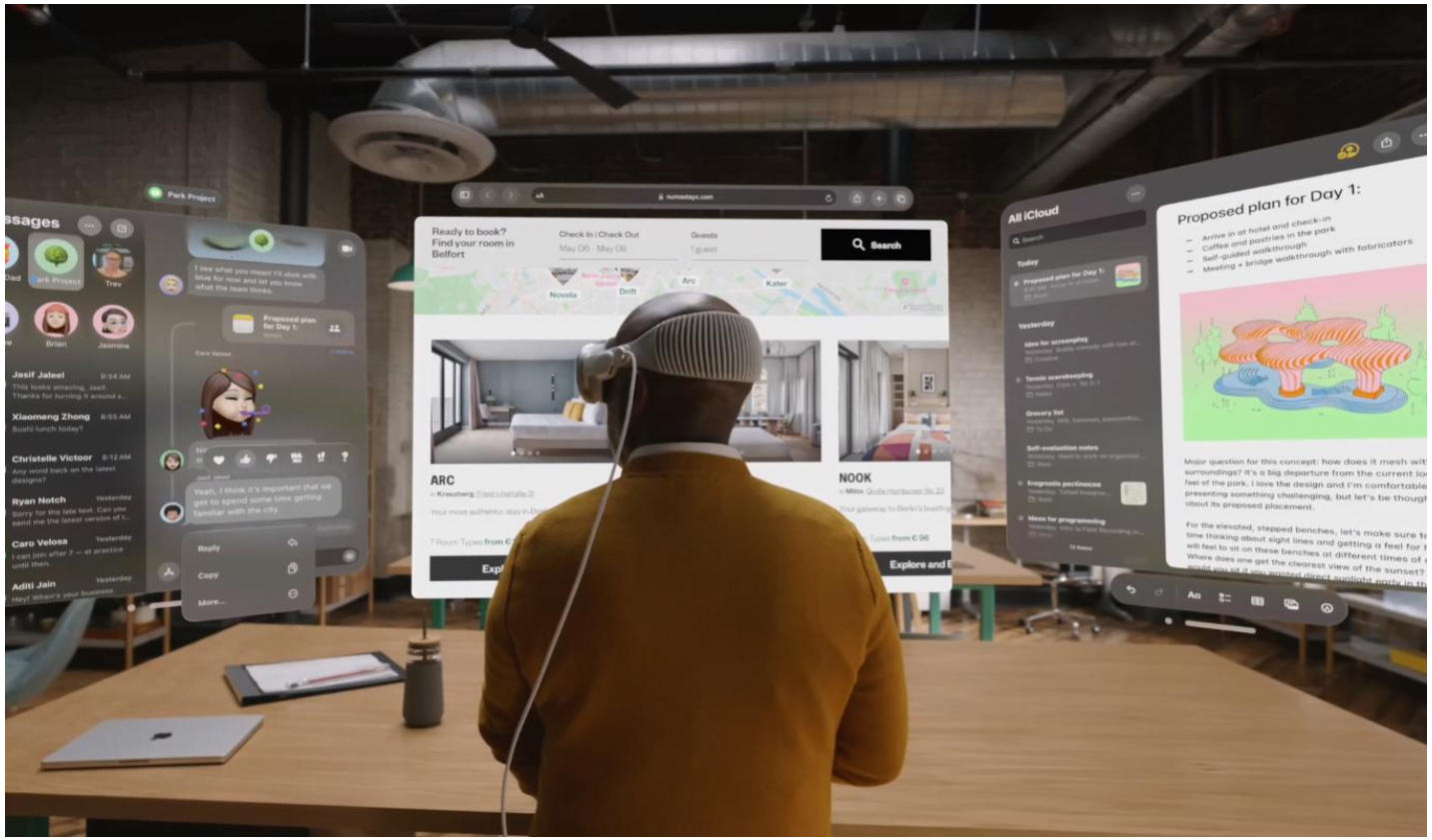
Improving the current framework to focus on website navigation and control



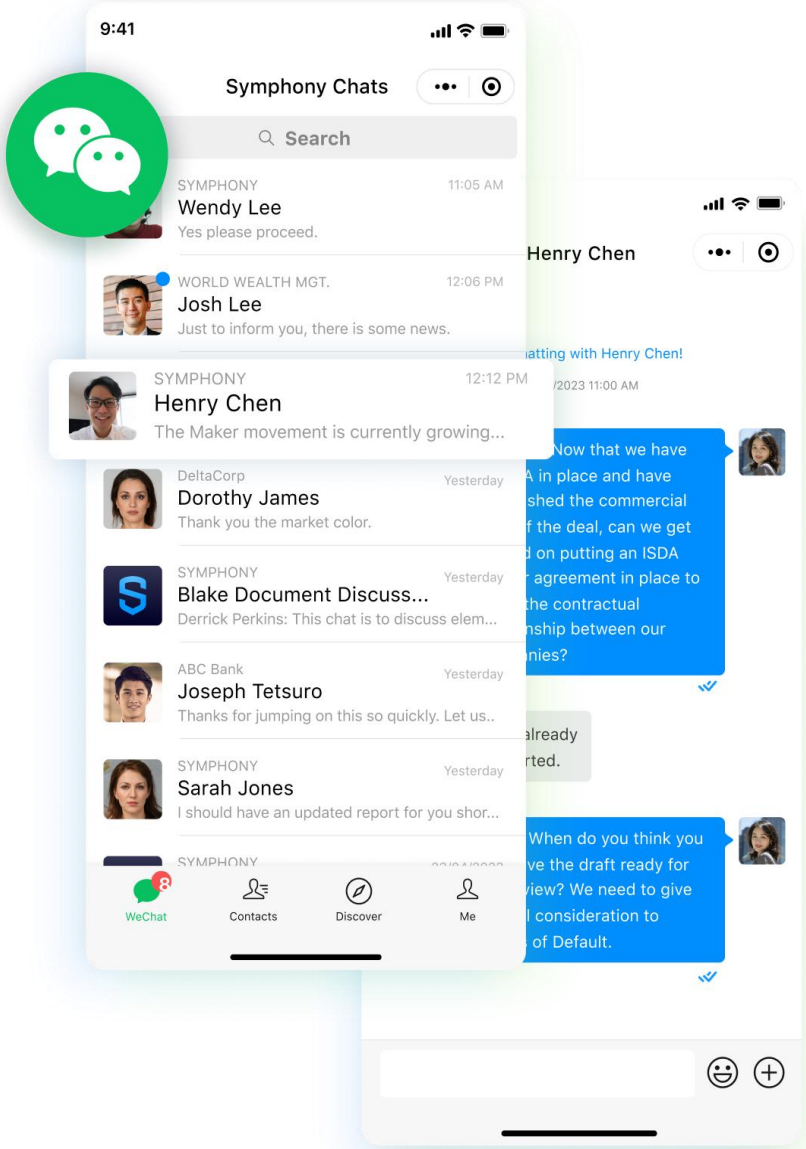
1. https://osu-nlp-group.github.io/Mind2Web/
2. https://www.multion.ai/

Exemplary Project 3: VR-based Agent

Extending the current framework to support VR platform (with voice control)



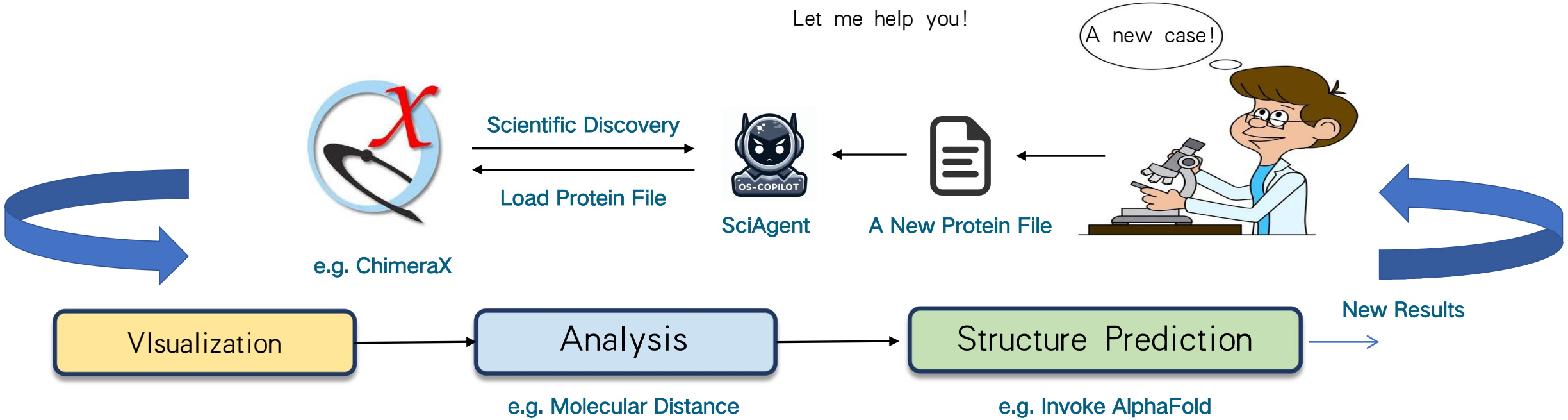
Exemplary Project 4: App-specific Agent

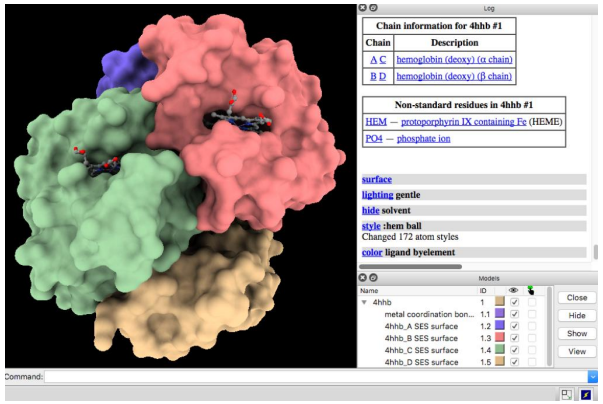


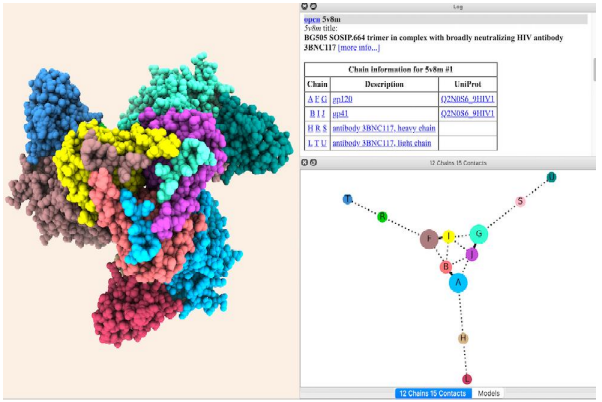
Excel

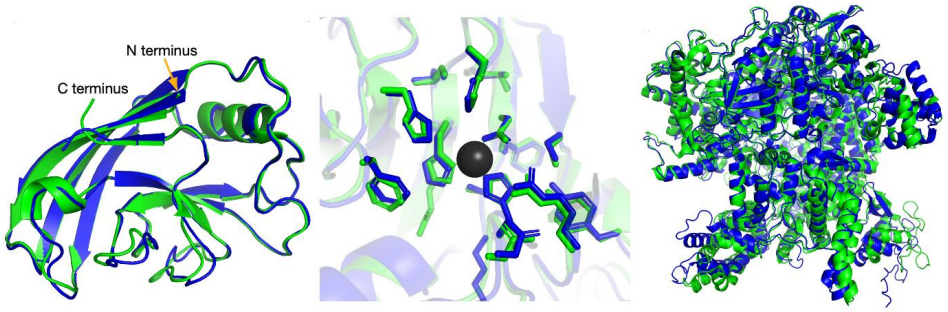
PowerPoint

Exemplary Project 5: Agent for Scientific Discovery



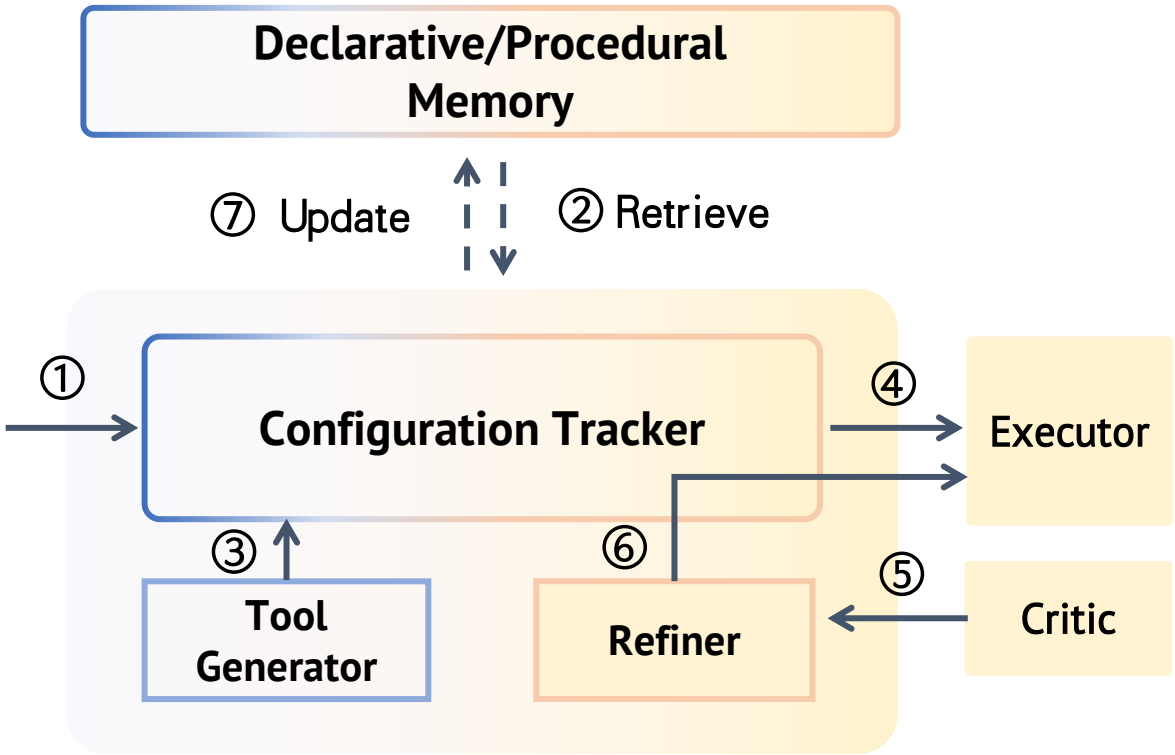






Exemplary Project 6: New Methodology

Support GUI interaction
Improve tool learning
Support more long-term memory
Personalization
Better planner
Better self-refine
....
Safety...
Interpretability...



More ....

Looking forward to exciting demonstrations from all of you!

Deliverables

1. Project proposal (0%):

Up to one page proposal contaning team information and brife introduction of the project

2. Presentation and demonstration (35%):

Present your project in the class and showcase your exciting demos!

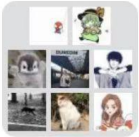
Evaluated based on excitment(15%), soundness(15%), and presentation clearness(5%).

3. Project Report (15%):

4-8 pages report concludes the project.

Resource

- OS-Copilot
  - Paper: https://arxiv.org/pdf/2402.07456
  - Code: https://github.com/OS-Copilot/OS-Copilot
  - Make PR/MR and become a contributor!
- Reading List:
  - released with the handout



群聊：COMP 7607 OS-Copilot 答疑群

该二维码7天内(9月15日前)有效，重新进入将更新