Plan-Point-Act: OLMo-Molmo Powered Web Agent

Federico Baldan
University of Washington
fbaldan@uw.edu

Long Cheng
University of Washington
lcheng97@uw.edu

Abstract

The project aims to develop an open-source web agent that integrates high-level action planning via models such as Qwen3 with AI2's fully open-source visual grounding model, Molmo, for precise UI interaction. By leveraging existing APIs from current LLMs and VLMs, the agent will reliably execute natural language commands through automated browser actions, eliminating the need for manual intervention. All implementation code will be publicly released to support open access and reproducibility.

1. Introduction

In this project, we aim to demonstrate that by leveraging off-the-shelf, open-source pretrained models for planning and visual grounding—without any additional training or large compute—we can build a fully functional, general-purpose web agent system. The system will be capable of executing tasks in a browser environment in an end-to-end fashion, relying only on open-source tools where possible.

The goal is to build an open-source web agent inspired by systems such as WebVoyager and BrowserGym agents, with components for planning, visual grounding, and automation. Our system will integrate pretrained open-source models such as Qwen3 [8] and OLMo [5], alongside specialized tools like Molmo [3] for grounding, EasyOCR [6], for text detection and GPT-4V-ACT [4] for DOM-based element extraction.

Our agentic workflow proceeds as follows:

1. The user inputs a natural language query. We post-process the input and query a pretrained planner model (e.g., Qwen3, OLMo) using prompt engineering and few-shot examples to generate a sequence of structured web actions.

2. Button labels and UI element descriptions are extracted from the predicted instructions. A visual grounding model (e.g., Molmo) or a DOM parser (e.g., GPT-4V-ACT) is used to locate the corresponding clickable regions.

3. Using an automation framework such as Playwright [1], the agent simulates interactions with the interface at the identified coordinates or DOM elements.

    - If the click is correct but the task is not yet completed, we loop back to step 2 to determine the next instruction.
    - If the click fails or the target is misidentified, we refine the grounding using fallback models or requery the planner.

Compared to agents using generic grounding or non-open-source components, our contribution lies in unifying open-source tools into a modular, extensible framework. While our system does not train an end-to-end model due to resource constraints, we focus on integrating key components, particularly the visual grounding and OCR modules.

1.1. Expected Results and Evaluation

We will measure task success rate and per-step accuracy on the BrowserGym [2]. / WorkArena [7] benchmark, aiming to surpass the 42.7% success rate reported for GPT-4 agents in the original WorkArena study. In addition, we plan to demonstrate the generalization capability of our agent across a wide range of tasks.

Acknowledgments

This project is completed as a requirement for Ranjay Krishna's course *Deep Learning for Computer Vision*, with mentorship from Zixian Ma and Ranjay Krishna. The project is conducted in collaboration with the Allen Institute for Artificial Intelligence (AI2), with support in model access and infrastructure.

References

[1] Playwright for python. https://playwright.dev/python/. 1

[2] Thibault Le Sellier De Chezelles, Maxime Gasse, Alexandre Drouin, Massimo Caccia, Léo Boisvert, Megh Thakkar, Tom Marty, Rim Assouel, Sahar Omidi Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, Dehan Kong, Frank F.

Xu, Siva Reddy, Quentin Cappart, Graham Neubig, Ruslan Salakhutdinov, Nicolas Chapados, and Alexandre Lacoste. The browsergym ecosystem for web agent research, 2025. 1

[3] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, Yen-Sung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models, 2024. 1

[4] Daniel Dupont. Gpt-4v-act, 2023. https://github.com/ddupont808/GPT-4V-Act. 1

[5] Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. Olmo: Accelerating the science of language models. *Preprint*, 2024. 1

[6] JaidedAI. Easyocr, 2020. https://github.com/JaidedAI/EasyOCR. 1

[7] ServiceNow Research. Workarena: How capable are web agents at solving common knowledge-work tasks?, 2024. Accessed: 1 May 2025. 1

[8] Qwen Team. Qwen3, April 2025. 1