



WALT: Web Agents that Learn Tools

Viraj Prabhu, Yutong Dai, Matthew Fernandez, Krithika Ramakrishnan, Jing Gu, Yanqi Luo, Silvio Savarese, Gaiming Xiong, Junnan Li, Zeyuan Chen, Ran Xu

<https://github.com/SalesforceAIResearch/WALT>



ICLR

Overview

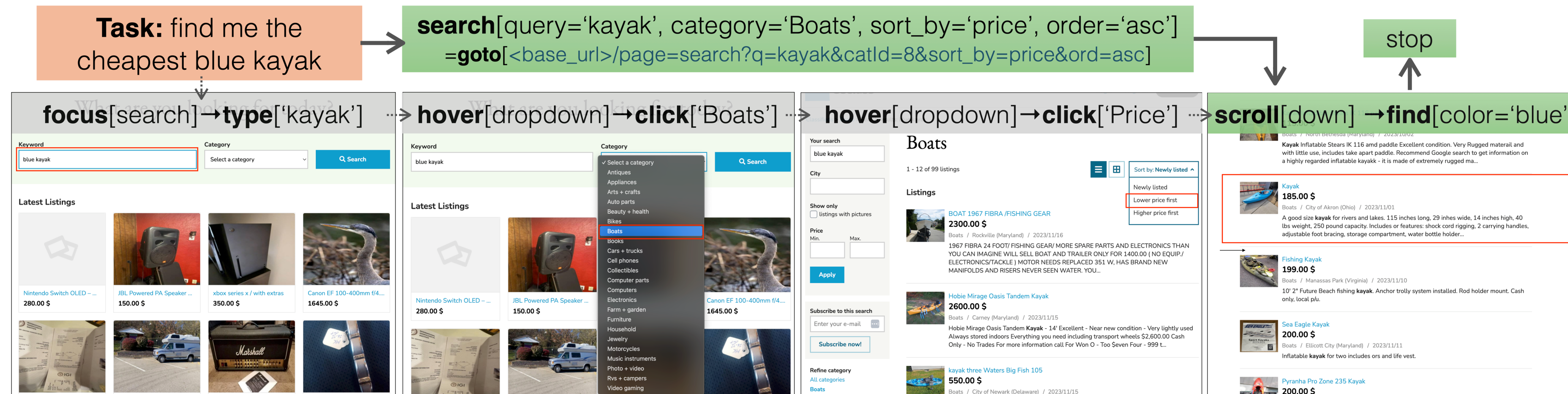
Problem: Modern web agents **break easily**, as they:

- i) **choreograph** clicks ii) **relearn** tools from scratch iii) are website **agnostic**

This **isn't** how humans browse the web! Instead, we:

- i) Identify **website-provided tools** ii) **Design** subtasks that map to these tools

Hypothesis: Web agents equipped w/ such tool-level macro-actions can achieve higher success rates in fewer steps



Traditional web agent

- 8+ fragile UI steps
- Heavy LLM reasoning

Web agents that Learn Tools (WALT)

- Single tool call
- 1 robust URL navigation (reverse-engineered!)

Contributions

- WALT reverse-engineers existing website functionality (e.g. search, filter, post, create) into **reusable callable tools**
- Tools **shift the agent's computational burden** away from low-level execution to high-level planning
- WALT achieves **SoTA success rates** on web agent benchmarks with **fewer avg. steps & LLM-calls**

Method

Propose → **[Demonstrate ↔ Generate ↔ Validate]**

I) **Tool Proposal:** browser agent systematically explores site sections, proposes **tool candidates** (goal, URL, elements)

II) **Tool Construction:** **demonstrate ↔ generate ↔ optimize ↔ validate**

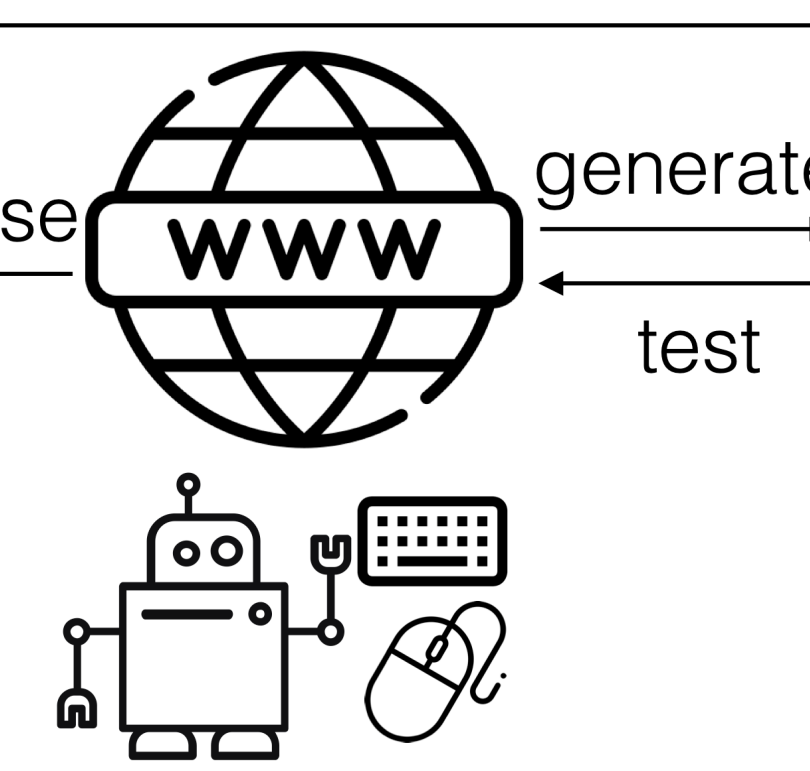
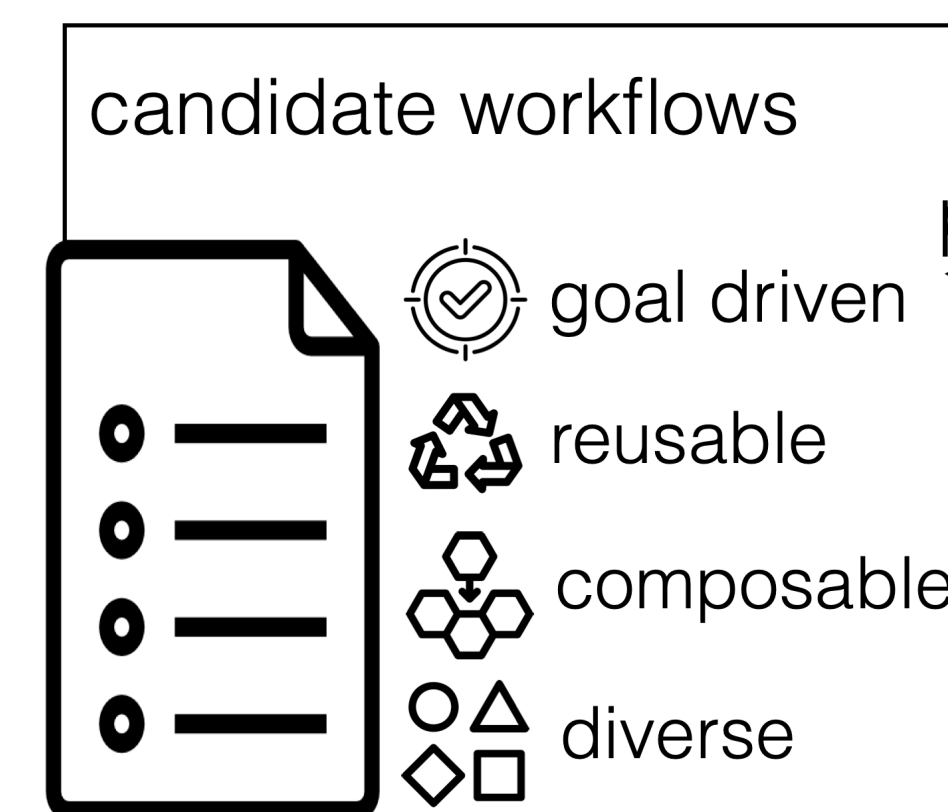
- Records execution trace; synthesizes input schema + action script
- URL reverse-engineering replaces UI sequences with deterministic navigation
- Stress-tested on pre-vetted inputs; failures trigger iterative refinement

$$\text{given } \tilde{u} = (s_i, E_i, G_i)$$

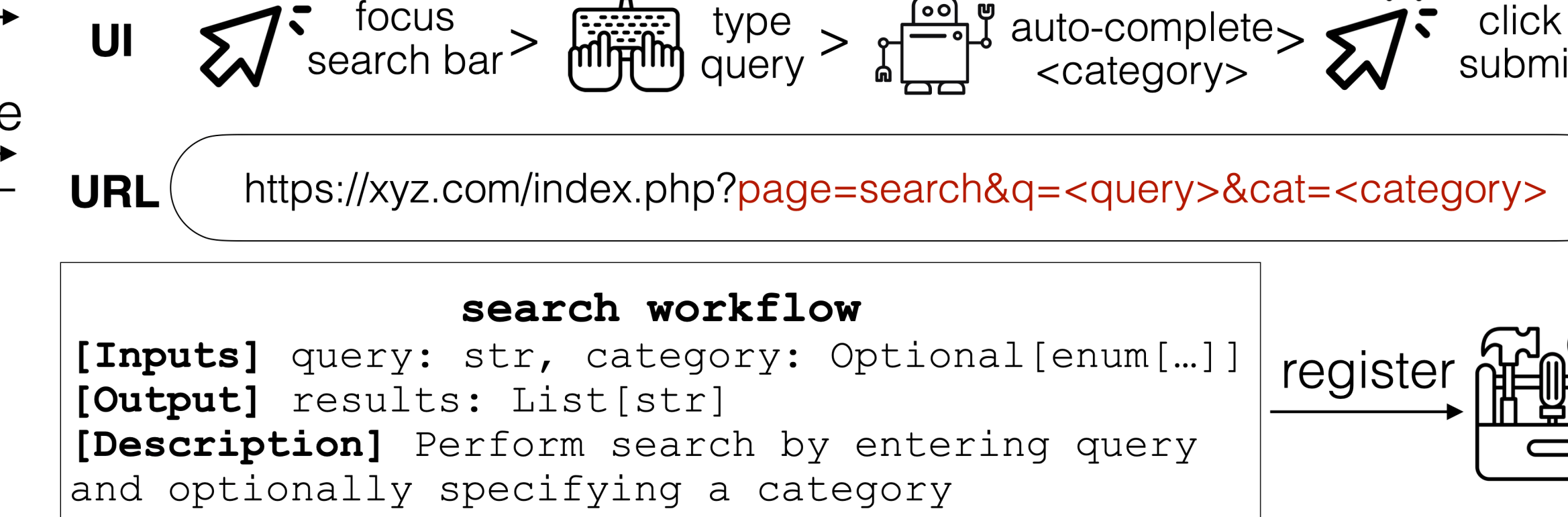
$$\text{execute \& generate } \tilde{u} \xrightarrow{\mathcal{B}_{\text{browser}}} \mathcal{X} \xrightarrow{\mathcal{B}_{\text{tool}}} (u, \mathcal{I}_{\text{test}})$$

$$\text{minimize FailRate}(u, \mathcal{I}_{\text{test}}) + \text{StepCount}(u) + \text{AgenticRatio}(u)$$

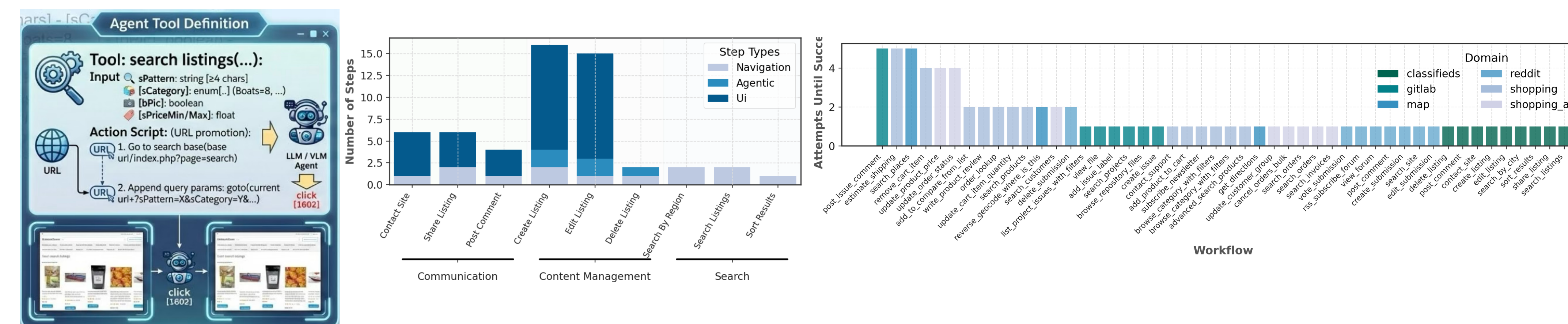
1. Strategic Exploration



2. Multi-level Exploitation

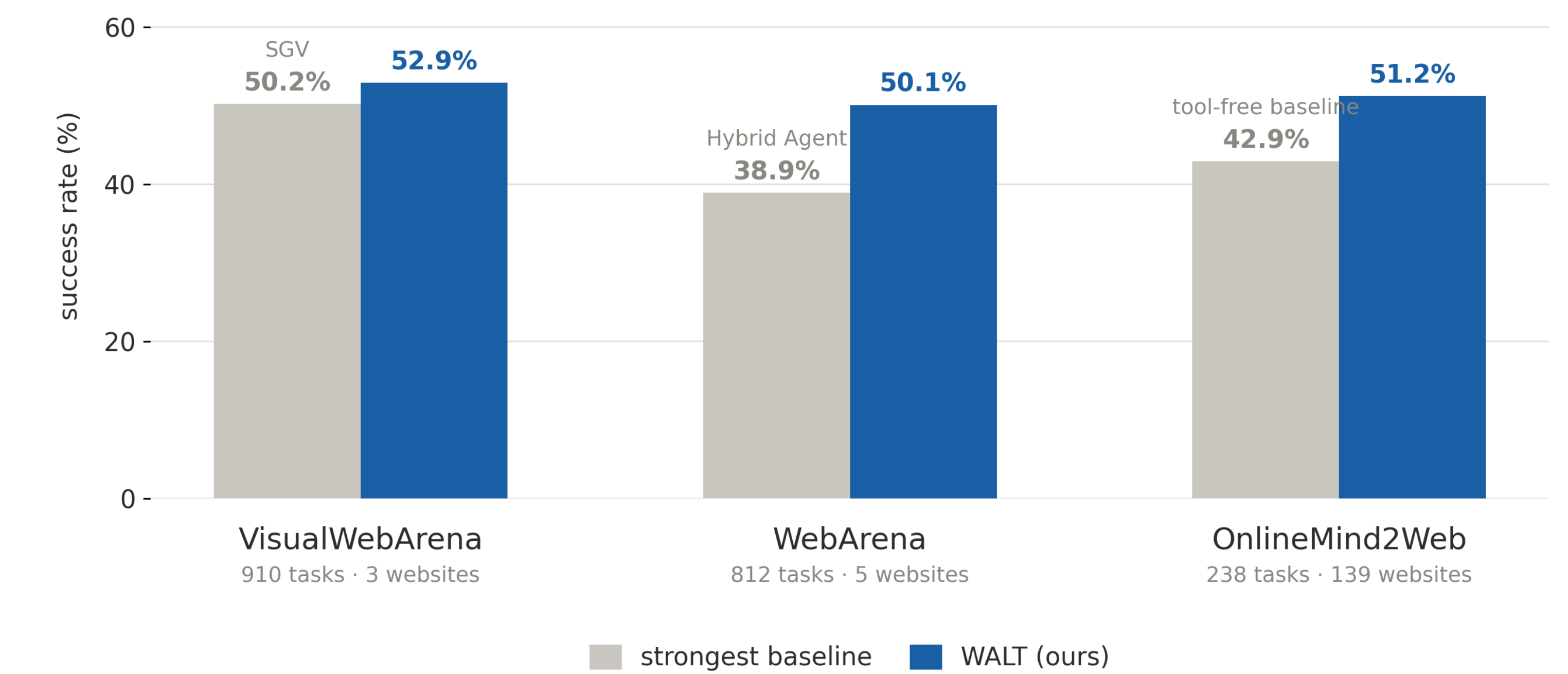


WALT learns >300 tools across diverse real & simulated websites (\$1.7 / tool)



Results

WALT achieves SoTA success rates with fewer LLM calls and no additional training



browser LLM	tools	dom-parser	verify	avg #steps (↓)	SR (%) ↑
gpt-4.1	none	text	self	7.6	34.9
gpt-4.1	discovered	text	self	6.6 _{-13.1%}	36.4 _{+4.3%}
gpt-5-mini	none	text	self	10.5	52.6
gpt-5-mini	discovered	text	self	8.3 _{-26.5%}	55.3 _{+5.1%}
gpt-5-mini	none	text	self	8.9	57.5
gpt-5-mini	discovered	text	self	6.5 _{-27.0%}	61.5 _{+7.0%}
gpt-5-mini	human demo	text	self	7.4 _{-16.0%}	66.0 _{+10.2%}
gpt-5-mini	none	multimodal	self	7.5 _{-15.7%}	59.0 _{+2.6%}
gpt-5-mini	none	text	external	11.0 _{-23.6%}	59.4 _{+3.3%}
gpt-5-mini	discovered	multimodal	external	7.0 _{-21.9%}	64.1 _{+11.5%}

