Breaking the Data Barrier - Building GUI Agents Through Task Generalization

打破数据壁垒——通过任务泛化构建 GUI 代理

Junlei Zhang* ♢I Zichen Ding *♠ Chang Ma* Zijie Chen ♢I Qiushi Sun* Zhenzhong Lan I Junxian He*

张俊磊 * ♢I 丁子辰 *♠ 马畅 * 陈子杰 ♢I 孙秋实 * 兰振中 I 何俊贤 *

♢ Zhejiang University I Westlake University ♠ Shanghai AI Laboratory

♢ 浙江大学 I 西湖大学 ♠ 上海人工智能实验室

* The University of Hong Kong ★ HKUST

* 香港大学 ★ 香港科技大学

Abstract

摘要

Graphical User Interface (GUI) agents offer cross-platform solutions for automating complex digital tasks, with significant potential to transform productivity workflows. However, their performance is often constrained by the scarcity of high-quality trajectory data. To address this limitation, we propose training Vision Language Models (VLMs) on data-rich, reasoning-intensive tasks during a dedicated mid-training stage, and then examine how incorporating these tasks in the mid-training phase facilitates generalization to GUI planning scenarios. Specifically, we explore a range of tasks with readily available instruction-tuning data, including GUI perception, multimodal reasoning, and textual reasoning. Through extensive experiments across 11 mid-training tasks, we demonstrate that: (1) Task generalization proves highly effective, yielding substantial improvements across most settings. For instance, multimodal mathematical reasoning enhances performance on AndroidWorld by an absolute 6.3%. Remarkably, text-only mathematical data significantly boosts GUI web agent performance, achieving a 5.6% improvement on WebArena and a 5.4% improvement on AndroidWorld, underscoring notable cross-modal generalization from text-based to visual domains; (2) Contrary to prior assumptions, GUI perception data-previously considered closely aligned with GUI agent tasks and widely utilized for training-has a comparatively limited impact on final performance; (3) Building on these insights, we identify the most effective mid-training tasks and curate optimized mixture datasets, resulting in absolute performance gains of 8.0% on WebArena and 12.2% on AndroidWorld. Our work provides valuable insights into cross-domain knowledge transfer for GUI agents and offers a practical approach to addressing data scarcity challenges in this emerging field. The code, data, and models are available at https://github.com/hkust-nlp/GUIMid.

图形用户界面 (GUI) 代理为自动化复杂数字任务提供了跨平台解决方案，具有显著改变生产力工作流程的潜力。然而，其性能常受限于高质量轨迹数据的稀缺性。为解决此限制，我们提出在专门的中期训练阶段，针对数据丰富且推理密集的任务训练视觉语言模型 (Vision Language Models, VLMs)，并探讨将这些任务纳入中期训练如何促进对 GUI 规划场景的泛化。具体而言，我们探索了一系列具有现成指令调优数据的任务，包括 GUI 感知、多模态推理和文本推理。通过对 11 个中期训练任务的广泛实验，我们证明:(1) 任务泛化极为有效，在大多数设置中带来显著提升。例如，多模态数学推理使 AndroidWorld 性能提升绝对值 6.3%。值得注意的是，纯文本数学数据显著提升了 GUI 网页代理性能，在 WebArena 上提升 5.6%，在 AndroidWorld 上提升 5.4%，凸显了从基于文本到视觉领域的显著跨模态泛化；(2) 与先前假设相反，先前被认为与 GUI 代理任务高度相关且广泛用于训练的 GUI 感知数据，对最终性能的影响相对有限；(3) 基于这些洞见，我们确定了最有效的中期训练任务并策划了优化的混合数据集，分别在 WebArena 和 AndroidWorld 上实现了绝对性能提升 8.0% 和 12.2%。我们的工作为 GUI 代理的跨领域知识迁移提供了宝贵见解，并为解决该新兴领域中的数据稀缺挑战提供了实用方法。代码、数据和模型可在 https://github.com/hkust-nlp/GUIMid 获取。

1 Introduction

1 引言

Interacting with graphical user interfaces (GUIs) has become a fundamental part of how humans engage with the world, from browsing the internet to using mobile apps. Developing autonomous agents capable of seamlessly interacting with these interfaces as personal assistants has the potential to revolutionize daily life (Xie et al., 2024; OpenAI, 2025; Wu et al., 2024), making it more efficient and convenient.

与图形用户界面 (GUI) 的交互已成为人类与世界互动的基本方式，从浏览互联网到使用移动应用。开发能够无缝与这些界面交互的自主代理，作为个人助理，有望彻底改变日常生活 (Xie et al., 2024；OpenAI, 2025；Wu et al., 2024)，使其更高效便捷。

Building GUI agents requires a combination of key capabilities: perception-understanding and interpreting GUI screenshots, grounding—translating human instructions into executable actions, and visual planning—carrying out tasks step by step to achieve the desired goal (Zheng et al., 2024b; Xu et al., 2024b; Ma et al., 2024). Among these, visual planning is the most challenging (Gur et al., 2023; Koh et al., 2024b; Yu et al., 2024). It demands breaking down complex instructions, like ”check my GitHub repositories with the most stars,” into

构建 GUI 代理需要结合多项关键能力: 感知——理解和解析 GUI 截图，落地——将人类指令转化为可执行操作，以及视觉规划——逐步执行任务以达成预期目标 (Zheng et al., 2024b；Xu et al., 2024b；Ma et al., 2024)。其中，视觉规划是最具挑战性的 (Gur et al., 2023；Koh et al., 2024b；Yu et al., 2024)。它要求将复杂指令，如“查看我 GitHub 上星标最多的仓库”，拆解为

---

*Co-first author. Work done during JZ’s visit to HKUST. Correspondance to Junlei Zhang (zhangjun-lei@westlake.edu.cn) and Junxian He (junxianh@cse.ust.hk).

* 共同第一作者。工作完成于张俊磊访问香港科技大学期间。通讯作者: 张俊磊 (zhangjunlei@westlake.edu.cn) 和何俊贤 (junxianh@cse.ust.hk)。

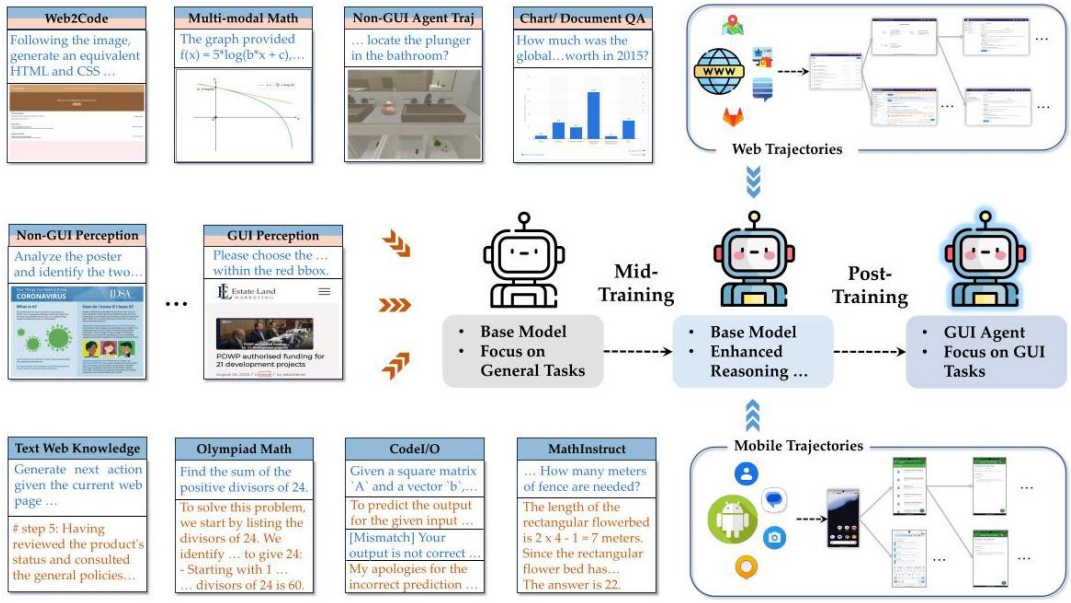

Figure 1: Overview of the mid-training and fine-tuning process. Left: We first train the GUI agent on mid-training data, primarily from non-GUI domains, to investigate whether the enhanced capabilities can generalize to GUI agent tasks; Right: We perform post-training on GUI trajectory data.

图 1: 中期训练与微调流程概览。左图: 我们首先在主要来自非 GUI 领域的中期训练数据上训练 GUI 代理，以探究增强能力是否能泛化至 GUI 代理任务；右图: 我们在 GUI 轨迹数据上进行后期训练。

precise, actionable steps. Vision-Language models (VLMs) naturally have the potential to serve as policy models for guiding GUI agent planning. With advanced prompting techniques, they can act as the foundational models for GUI agents (Zheng et al., 2024b; Song et al., 2024). However, most existing VLMs lack the reliability and stability needed to perform as effective GUI agents, often delivering subpar performance on benchmarks (Zhou et al., 2023; Deng et al., 2023; Koh et al., 2024a; Xie et al., 2024). For example, gpt4-o (Hurst et al., 2024) achieves a mere 15.6% on WebArena (Zhou et al., 2023), and a Qwen2-VL-7B-Instruct (Wang et al., 2024a) could barely generate goal-aligned actions. Most errors stem from insufficient planning when tasks require multiple sequential steps. To address this limitation, researchers have been trying to improve the GUI policy VLMs by collecting or synthesizing GUI trajectory data (Xu et al., 2024b;a; Sun et al., 2024b; Ou et al., 2024). However, real-world GUI trajectory data is not readily available, making the acquisition of diverse and high-quality GUI datasets a significant challenge. Additionally, synthesizing trajectories using large models often results in low-quality outputs, as even the most advanced VLMs struggle to perform effectively on realistic GUI agent tasks. In light of this challenge, we focus our study on the critical question: How to improve the agentic abilities of VLMs for GUI tasks with more scalable data sources?

精确且可执行的步骤。视觉-语言模型 (Vision-Language Models，VLMs) 天然具备作为策略模型指导 GUI 代理规划的潜力。借助先进的提示技术，它们可以作为 GUI 代理的基础模型 (Zheng et al., 2024b；Song et al., 2024)。然而，大多数现有的 VLMs 缺乏作为有效 GUI 代理所需的可靠性和稳定性，常在基准测试中表现不佳 (Zhou et al., 2023；Deng et al., 2023；Koh et al., 2024a；Xie et al., 2024)。例如，gpt4-o(Hurst et al., 2024) 在 WebArena(Zhou et al., 2023) 上的得分仅为 15.6%，而 Qwen2-VL-7B-Instruct(Wang et al., 2024a) 几乎无法生成与目标一致的动作。大多数错误源于任务需要多步连续操作时规划不足。为解决此限制，研究者尝试通过收集或合成 GUI 轨迹数据来改进 GUI 策略 VLMs(Xu et al., 2024b;a；Sun et al., 2024b；Ou et al., 2024)。然而，现实世界的 GUI 轨迹数据难以获得，使得多样且高质量的 GUI 数据集采集成为重大挑战。此外，利用大型模型合成轨迹往往导致输出质量低下，因为即使是最先进的 VLMs 也难以在真实 GUI 代理任务中有效表现。鉴于此挑战，我们的研究聚焦于关键问题: 如何利用更具扩展性的数据源提升 VLMs 在 GUI 任务中的代理能力?

To this end, we introduce a mid-training stage to enhance the foundational agentic capabilities of VLMs prior to fine-tuning them on a limited subset of GUI trajectories for task-specific adaptation. Mid-training (illustrated in Figure 1) refers to an intermediate training stage between pre-training and fine-tuning, where models are enhanced with specialized capabilities for better adaptation. [1] While this approach has been successfully applied in previous studies across various scenarios involving LLMs and VLMs (Wang et al., 2024b; Sun et al., 2024a; Yang et al., 2025), it remains unclear which types of tasks can be effectively generalized to learning GUI agents, given the complexity and typically extensive context associated with such tasks. Previous research exploring alternative data sources has primarily focused on GUI-specific resources, such as web tutorials and GUI captions (Chen et al., 2024; Xu et al., 2024a; Ou et al., 2024), which falls short on representing

为此，我们引入了一个中间训练阶段，以增强 VLMs 的基础代理能力，然后再在有限的 GUI 轨迹子集上进行微调，实现任务特定的适应。中间训练 (如图 1 所示) 指的是介于预训练和微调之间的一个中间训练阶段，通过该阶段模型获得专门能力以更好地适应任务。[1] 虽然该方法已在先前涉及大型语言模型 (LLMs) 和视觉语言模型 (VLMs) 的多种场景中成功应用 (Wang et al., 2024b；Sun et al., 2024a；Yang et al., 2025)，但鉴于 GUI 任务的复杂性及其通常涉及的大量上下文，目前尚不清楚哪些类型的任务能有效泛化至 GUI 代理学习。此前探索替代数据源的研究主要聚焦于 GUI 特定资源，如网页教程和 GUI 字幕 (Chen et al., 2024；Xu et al., 2024a；Ou et al., 2024)，这在表现

---

[1] The definition of mid-training can differ, conceptually it is similar to continual pretraining on instruction tuning data in our context.

[1] 中间训练的定义可能有所不同，概念上在我们的语境中类似于在指令调优数据上的持续预训练。

---

agentic planning abilities. In this work, we investigate a diverse set of mid-training data domains to evaluate their impact on learning GUI agents. These domains include general image perception, chart understanding, multimodal reasoning, and text-based tasks such as mathematics and programming.

代理规划能力。在本工作中，我们考察了一组多样的中间训练数据领域，以评估它们对学习 GUI 代理的影响。这些领域包括通用图像感知、图表理解、多模态推理以及基于文本的任务，如数学和编程。

We evaluate eleven datasets—seven multimodal and four textual—focusing on reasoning, knowledge retrieval, and perception (§3.1). Samples are collected per domain, followed by separate mid-training on each, and fine-tuning on a GUI trajectory dataset. By standardizing all datasets to generate high-level action thoughts before grounding to actions, we enhance planning transfer. A continuous optimizer ensures smooth training transitions and minimizes forgetting. Analysis on mobile and web benchmarks validates the effectiveness of our mid-training approach. Pure text mathematical datasets show the largest gains, improving AndroidWorld by 5.4% and WebArena by 5.6%, demonstrating reasoning abilities could transfer cross-domain. Coding datasets boost performance by around 3.0% on both tasks. Surprisingly, visual perception datasets yield modest gains, likely due to existing VLMs' strong visual capabilities. Based on these insights, we introduce GUIMid, a 300k dataset combining the four best-performing domains. GUIMid achieves SOTA on AndroidWorld in pure-visual settings and improves Qwen2-VL to GPT4-o level performances on web browsing, with overall gains of 12.2% and 8.0% on AndroidWorld and WebArena.

我们评估了十一组数据集——七个多模态和四个文本，重点关注推理、知识检索和感知 (§3.1)。每个领域收集样本，随后分别进行中间训练，再在 GUI 轨迹数据集上微调。通过统一所有数据集生成高级动作思考再落地为动作，我们增强了规划的迁移能力。连续优化器确保训练平滑过渡并最小化遗忘。移动端和网页基准测试的分析验证了我们中间训练方法的有效性。纯文本数学数据集带来最大提升，AndroidWorld 提升 5.4%，WebArena 提升 5.6%，表明推理能力可跨领域迁移。编程数据集在两项任务上均提升约 3.0%。令人惊讶的是，视觉感知数据集带来较小提升，可能因现有 VLMs 已具备较强视觉能力。基于这些洞见，我们推出了 GUIMid，一个包含四个表现最佳领域的 30 万条数据集。GUIMid 在纯视觉设置下实现了 AndroidWorld 的最新水平 (SOTA)，并使 Qwen2-VL 在网页浏览任务上达到 GPT4-o 水平，AndroidWorld 和 WebArena 整体提升分别为 12.2% 和 8.0%。

2 Vision-based GUI Agent Framework

2 基于视觉的 GUI 代理框架

Pure vision-based GUI agents. Previous work (Gur et al., 2023; Zheng et al., 2024b; Xie et al., 2024; Zhou et al., 2023) has largely relied on structural text-based GUI representations, such as HTML or accessibility trees. In contrast, we focus on the more challenging pure-vision setting, where vision-based agents take screenshots and task descriptions as input, generating coordinate-based actions directly within pixel space. This pure-vision approach provides key advantages: (1) it eliminates dependencies on backend structures, enabling cross-platform operation while avoiding the noise often present in accessibility trees, and (2) it aligns more closely with human interaction patterns, allowing for seamless integration into real-world workflows.

纯视觉驱动的 GUI 代理。以往的工作 (Gur 等，2023；Zheng 等，2024b；Xie 等，2024；Zhou 等，2023) 主要依赖于基于结构化文本的 GUI 表示，如 HTML 或辅助功能树。相比之下，我们聚焦于更具挑战性的纯视觉场景，其中基于视觉的代理以截图和任务描述作为输入，直接在像素空间内生成基于坐标的操作。这种纯视觉方法具有关键优势:(1) 消除了对后端结构的依赖，实现跨平台操作，同时避免了辅助功能树中常见的噪声；(2) 更贴合人类交互模式，便于无缝融入真实工作流程。

The inference process can be formalized as a specialized instance of Partially Observable Markov Decision Processes (POMDPs), represented by tuple $\langle g, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T} \rangle$, where g denotes the task goal, $\mathcal{S}$ the state space, $\mathcal{A}$ the action space, $\widetilde{\mathcal{O}}$ the observation space (visual feedback from the screen), and $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ the state transition

function. At each time step t , the agent executes decisions according to policy π , which integrates the task goal g , memory $m_t = \{o_j, a_j, o_{j+1}, a_{j+1}, \ldots, o_{t-1}, a_{t-1}\}, 0 \leq j < t$ (capturing the history of actions and observations), and the current observation o_t . The agent' s trajectory, denoted as $\tau = [s_0, a_0, s_1, a_1, \ldots, s_t]$, emerges from the policy and environmental state transitions, as formulated by:

推理过程可以形式化为部分可观测马尔可夫决策过程 (Partially Observable Markov Decision Processes，POMDPs) 的特例，由元组 $\langle g, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T} \rangle$ 表示，其中 g 表示任务目标，$\mathcal{S}$ 表示状态空间，$\mathcal{A}$ 表示动作空间，$\widetilde{\mathcal{O}}$ 表示观测空间 (来自屏幕的视觉反馈)，$\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ 表示状态转移函数。在每个时间步 t ，代理根据策略 π 执行决策，该策略整合了任务目标 g 、记忆 $m_t = \{o_j, a_j, o_{j+1}, a_{j+1}, \ldots, o_{t-1}, a_{t-1}\}, 0 \leq j < t$ (捕捉动作和观测的历史) 以及当前观测 o_t 。代理的轨迹，记为 $\tau = [s_0, a_0, s_1, a_1, \ldots, s_t]$ ，由策略和环境状态转移共同决定，其公式为:

$$p_\pi(\tau) = p(s_0) \prod_{t=0}^{T} \pi(a_t \mid g, s_t, m_t) \mathcal{T}(s_{t+1} \mid s_t, a_t) \tag{1}$$

We then introduce specific implementations of the observation and action space in our vision-based GUI agent framework.

接下来，我们介绍在基于视觉的 GUI 代理框架中观测空间和动作空间的具体实现。

The action space. Our GUI agent employs a coordinate-based action space to ensure cross-platform compatibility and realistic human-like interactions. The policy model generating actions consists of two components: a planner model and a grounding model. The planner model first generates high-level action description following a planning-rich thought, i.e. "Task instruction <thought> <high-level action>". Then the grounding model maps the high-level action into screenshot manipulations.

动作空间。我们的 GUI 代理采用基于坐标的动作空间，以确保跨平台兼容性和逼真的类人交互。生成动作的策略模型由两个部分组成: 规划模型和落地模型。规划模型首先生成遵循丰富规划思路的高层动作描述，即“任务指令 <thought> < 高层动作 >”。然后，落地模型将高层动作映射为截图上的具体操作。

For instance, to search for "GUI agents" in a search engine, the generated content is:

例如，要在搜索引擎中搜索“GUI 代理”，生成的内容为:

<thought>: To find unlabeled issues in the metaseq GitLab repository, click the "Issues" tab in the main navigation menu, then filter for issues without labels.

<thought>: 要在 metaseq GitLab 仓库中查找未标记的问题，点击主导航菜单中的“Issues”标签，然后筛选无标签的问题。

Domains	Ability	Datasets	Samples	Type
Vision-and-Language Modality				
Chart/Document QA	Perception	InfographicVQA (Guo et al., 2024)	2,184	Instruction, Thought", Answer
		Ureader QA (Guo et al., 2024)	53,794	Instruction, Thought, Answer
		MPDocVQA (Tito et al., 2023)	431	Instruction, Thought, Answer
		MathV360k (Liu et al., 2024b)	93,591	Instruction, Thought, Answer
Non-GUI Perception	Perception	Ureader OCR (Ye et al., 2023)	6,146	Instruction, Thought', Answer
		DUE (Borchmann et al., 2021)	143,854	Instruction, Answer
GUI Perception	Perception	MultiUI (Liu et al., 2024a)	150,000	Instruction, Answer
Web Screenshot2Code	Perception	Web2Code (Yun et al., 2024)	150,000	Instruction, Answer
Multi-modal Math	Reasoning	Mavis (Zhang et al., 2024b)	150,000	Instruction, Thought, Answer
Multi-round Visual Conversation	Interaction	SVIT (Zhao et al., 2023)	150,000	Instruction, Thought, Answer
Non-GUI Agent Trajectories	Interaction	AlfWorld (Guo et al., 2024)	51,780	Instruction, Thought, Answer
Language Modality				
MathInstruct	Reasoning	MathInstruct (Yue et al., 2023)	150,000	Instruction, Thought, Answer
Olympiad Math	Reasoning	NuminaMath (LI et al., 2024)	150,000	Instruction, Thought, Answer
CodeI/O	Reasoning	CodeI/O (Li et al., 2025)	150,000	Instruction, Thought, Answer
Web Knowledge Base	Knowledge	Synatra (Ou et al., 2024)	99,924	Instruction, Thought, Answer
		AgentTrek (Xu et al., 2024a)	50,076	Instruction, Thought, Answer

领域	能力	数据集	样本	类型
视觉-语言模态				
图表/文档问答	感知	InfographicVQA (Guo et al., 2024)	2,184	指令，思考，答案
		Ureader 问答 (Guo et al., 2024)	53,794	指令，思考，答案
		MPDocVQA (Tito et al., 2023)	431	指令，思考，答案
		MathV360k (Liu et al., 2024b)	93,591	指令，思考，答案
非图形界面感知	感知	Ureader OCR (Ye et al., 2023)	6,146	指令，思考，答案
		DUE (Borchmann et al., 2021)	143,854	指令，答案
图形界面感知	感知	MultiUI (Liu et al., 2024a)	150,000	指令，答案
网页截图转代码	感知	Web2Code (Yun et al., 2024)	150,000	指令，答案
多模态数学	推理	Mavis (Zhang et al., 2024b)	150,000	指令，思考，答案
多轮视觉对话	交互	SVIT (Zhao et al., 2023)	150,000	指令，思考，答案
非图形界面代理轨迹	交互	AlfWorld (Guo et al., 2024)	51,780	指令，思考，答案
语言模态				
MathInstruct	推理	MathInstruct (Yue et al., 2023)	150,000	指令，思考，答案
奥林匹克数学	推理	NuminaMath (LI et al., 2024)	150,000	指令，思考，答案
代码输入/输出	推理	CodeI/O (Li et al., 2025)	150,000	指令，思考，答案
网络知识库	知识	Synatra (Ou et al., 2024)	99,924	指令，思考，答案
		AgentTrek (Xu et al., 2024a)	50,076	指令，思考，答案

Table 1: Statistics of the domains and corresponding datasets used in mid-training, (*) indicates that some instructions in the dataset do not require a "Thought" (e.g., "Answer concisely with one word or phrase.").

表 1: 中期训练中使用的领域及对应数据集的统计，(*) 表示数据集中部分指令不需要“思考”(例如，“用一个词或短语简洁回答。”)。

<high-level action>:

<high-level action>:

{
”Element Description”: ”Click the Issues tab in the main navigation menu”,

“元素描述”：“点击主导航菜单中的问题 (Issues) 标签”,

”Action”: ”click”,

“操作”：“点击”,

}
<grounded action>: Click [coordinate_x 0.12] [coordinate_y 0.07]

<grounded action>: 点击 [coordinate_x 0.12] [coordinate_y 0.07]

We use UGround-V1-7B (Gou et al., 2025) for grounding, and our trained policy model for thought and high-level action generation. This separation also enables better transfer of planning ability through mid-training (as detailed in §3) in addition to flexible inclusion of new actions. The details of action spaces for mobile and web tasks can be found in Table 6, 7.

我们使用 UGround-V1-7B(Gou 等，2025) 进行定位，使用我们训练的策略模型生成思考和高层次动作。这种分离还使得通过中期训练 (详见 §3) 更好地传递规划能力成为可能，同时灵活地包含新动作。移动和网页任务的动作空间详情见表 6、7。

The observation space and memory. The observation space is visual-rich in our framework, comprising of a screenshot of the current screen and simple meta-information (i.e., the current URL for web tasks). To augment agent memory, we also provide the model with a history of previous actions, using the concatenated output of planner generated high-level actions, e.g. ”step 1: click ’the search results titled with wikipedia’; step 2: type ’GUI Agent’ into the search bar at the top of the page”.

观察空间和记忆。在我们的框架中，观察空间视觉信息丰富，包括当前屏幕截图和简单的元信息 (即网页任务的当前 URL)。为了增强代理记忆，我们还向模型提供了之前动作的历史，使用规划器生成的高层次动作的串联输出，例如“步骤 1: 点击‘标题为维基百科的搜索结果’；步骤 2: 在页面顶部的搜索栏输入‘GUI Agent’”。

3 Breaking the Data Barrier via Mid-Training

3 通过中期训练突破数据瓶颈

Despite advancements in vision-language models, GUI agent training faces challenges due to limited high-quality trajectory data. We introduce a mid-training stage between general pre-training and task-specific post-training. This approach leverages abundant data from adjacent domains—image perception, chart understanding,

multimodal reasoning, and programming—to develop foundational capabilities before GUI-specific adaptation. Our two-step strategy includes mid-training on scalable data sources (§3.1) followed by fine-tuning on a small GUI trajectory dataset (§3.2). Our optimized training procedure introduced in §3.3 ensures consistent generalization to GUI agent skills. We briefly introduce these datasets and the training procedure in this section, and more details can be found in Appendix B, C.

尽管视觉语言模型取得了进展，GUI 代理训练仍面临高质量轨迹数据有限的挑战。我们引入了一个介于通用预训练和任务特定后训练之间的中期训练阶段。该方法利用来自相邻领域——图像感知、图表理解、多模态推理和编程——的大量数据，在 GUI 特定适应之前培养基础能力。我们的两步策略包括在可扩展数据源上的中期训练 (§3.1)，随后在小规模 GUI 轨迹数据集上的微调 (§3.2)。我们在 §3.3 介绍的优化训练流程确保了对 GUI 代理技能的一致泛化。本文简要介绍这些数据集和训练流程，更多细节见附录 B、C。

3.1 Mid-Training Data

3.1 中期训练数据

We collect 150k diverse training samples for each domain to study cross-domain generalization. For the non-GUI agent domain, we included 51k samples due to the scarcity of

我们为每个领域收集了 15 万条多样化训练样本以研究跨领域泛化。由于非 GUI 代理领域样本稀缺，我们仅包含了 5.1 万条样本。

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

领域	数据集	样本	类型
网页	OS-Genesis(网页)(Sun 等，2024b)	3,789	指令、思考、行动
	MM-Mind2Web(Zheng 等，2024a)	21,542	指令、思考、行动
	VisualWebArena(Koh 等，2024a)	3,264	指令、思考、行动
移动端	OS-Genesis(移动端)(Sun 等，2024b)	4,941	指令、思考、行动
	Aguvis(Xu 等，2024b)	22,526	指令、思考、行动

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

表 2: 用于后训练的网页/移动端领域及相应 GUI 轨迹数据集统计。

agent trajectories. The mid-training domains are listed in Table 1. For vision-language tasks, we include: Chart/Document QA (Guo et al., 2024; Tito et al., 2023) and Multi-modal Math (Zhang et al., 2024b) for fine-grained understanding and visual reasoning; Non-GUI Perception tasks (Ye et al., 2023; Borchmann et al., 2021)

including Document OCR for fundamental visual understanding; Web Screenshot2Code (Yun et al., 2024) for structured web screenshot interpretation; and multi-turn data from Visual Conversations (Zhao et al., 2023) and Non-GUI Agent Trajectories (Guo et al., 2024) to enhance VLM interactive capabilities.

代理轨迹。中期训练领域列于表 1。对于视觉-语言任务，我们包括: 图表/文档问答 (Guo 等，2024；Tito 等，2023) 和多模态数学 (Zhang 等，2024b) 以实现细粒度理解和视觉推理；非 GUI 感知任务 (Ye 等，2023；Borchmann 等，2021)，包括文档 OCR 以实现基础视觉理解；网页截图转代码 (Yun 等，2024) 用于结构化网页截图解析；以及来自视觉对话 (Zhao 等，2023) 和非 GUI 代理轨迹 (Guo 等，2024) 的多轮数据，以增强视觉语言模型 (VLM) 的交互能力。

Complementing these, we include pure-text data featuring more reasoning-intensive, and knowledge-rich tasks, including from medium-difficulty (MathInstruct (Yue et al., 2023)) to challenging (Olympiad Math (LI et al., 2024)) mathematical reasoning, Code I/O (Li et al., 2025) to develop procedural reasoning through code generation, and Web Knowledge Base (Ou et al., 2024; Xu et al., 2024a) to inject domain knowledge.

作为补充，我们包含纯文本数据，涵盖更多推理密集且知识丰富的任务，范围从中等难度 (MathInstruct(Yue 等，2023)) 到挑战性较高的 (奥林匹克数学 (LI 等，2024)) 数学推理，代码输入输出 (Code I/O)(Li 等，2025) 以通过代码生成发展程序化推理，以及网络知识库 (Ou 等，2024；Xu 等，2024a) 以注入领域知识。

3.2 Post-Training GUI Trajectory Data

3.2 后训练 GUI 轨迹数据

For post-training, we used high-quality GUI trajectories from state-of-the-art systems across platforms. We incorporated web and mobile data from OS-Genesis (Sun et al., 2024b), Aguvis (Xu et al., 2024b), and MM-Mind2Web (Zheng et al., 2024a), plus 3.2K manually annotated steps from VisualWebArena (Koh et al., 2024a). Our final dataset contains 28K web samples and 27K mobile samples, as detailed in Table 2.

在后训练阶段，我们使用了来自跨平台最先进系统的高质量 GUI 轨迹。我们整合了来自 OS-Genesis(Sun 等，2024b)、Aguvis(Xu 等，2024b) 和 MM-Mind2Web(Zheng 等，2024a) 的网页和移动端数据，以及 VisualWebArena(Koh 等，2024a) 中 3.2 千条手工标注步骤。我们的最终数据集包含 2.8 万条网页样本和 2.7 万条移动端样本，详见表 2。

3.3 Training Method

3.3 训练方法

We employ a two-stage training approach consisting of mid-training followed by fine-tuning on GUI trajectory data. Both stages are integrated under a single optimizer and learning rate schedule. During the mid-training stage, we mix the post-training GUI trajectory data into the mid-training domain data (e.g. ChartQA) to mitigate potential forgetting issues, which we will analyze empirically in §4.4. This mixing is particularly crucial when there exists a substantial domain gap between the mid-training data and GUI trajectory data, especially when the former is

primarily text-based (see Figure 4 for comparison). Additional training hyperparameters and details are detailed in Appendix D.

我们采用两阶段训练方法，先进行中期训练，随后在 GUI 轨迹数据上微调。两个阶段均在统一的优化器和学习率调度下进行。在中期训练阶段，我们将后训练的 GUI 轨迹数据混入中期训练领域数据 (如 ChartQA)，以缓解潜在的遗忘问题，相关分析见 §4.4。当中期训练数据与 GUI 轨迹数据存在显著领域差异，尤其是前者主要为文本时 (参见图 4 对比)，此混合尤为关键。更多训练超参数和细节详见附录 D。

4 Experiments

4 实验

4.1 Experimental Setup

4.1 实验设置

To explore how non-GUI trajectory data can enhance VLMs' foundational agentic capabilities, we study 7 multi-modal domains and 4 language domains (Table 1). For the post-training data, we collect 56K high-quality data (Table 2). Following State-Of-The-Art (SOTA) works (Xu et al., 2024b; Sun et al., 2024b; Gou et al., 2025), we employ Qwen2-VL-7B-Instruct (Wang et al., 2024a) as our backbone. We first train our model on the mid-training datasets, then fine-tune it on GUI trajectories. To study whether these separate domains can be combined to achieve superior performance, we randomly sample a total of 300K examples from high-performing domains (150K from MathInstruct, 20K from CodeI/O, 50K from Olympiads Math, and 80K from Multi-modal Math) to create a consolidated mid-training dataset called GUIMid. During the mid-training stage, we mix the GUI trajectory samples into our mid-training dataset to make the training stable.

为探究非 GUI 轨迹数据如何提升视觉语言模型 (VLM) 的基础代理能力，我们研究了 7 个多模态领域和 4 个语言领域 (表 1)。后训练数据收集了 5.6 万条高质量数据 (表 2)。遵循最新研究 (Xu 等，2024b；Sun 等，2024b；Gou 等，2025)，我们采用 Qwen2-VL-7B-Instruct(Wang 等，2024a) 作为骨干模型。先在中期训练数据集上训练模型，再在 GUI 轨迹上微调。为研究这些不同领域数据是否能合并以实现更优性能，我们从高性能领域随机采样共 30 万条样本 (15 万来自 MathInstruct，2 万来自 CodeI/O，5 万来自奥林匹克数学，8 万来自多模态数学)，构建了名为 GUIMid 的合并中期训练数据集。中期训练阶段，我们将 GUI 轨迹样本混入中期训练数据集以保持训练稳定。

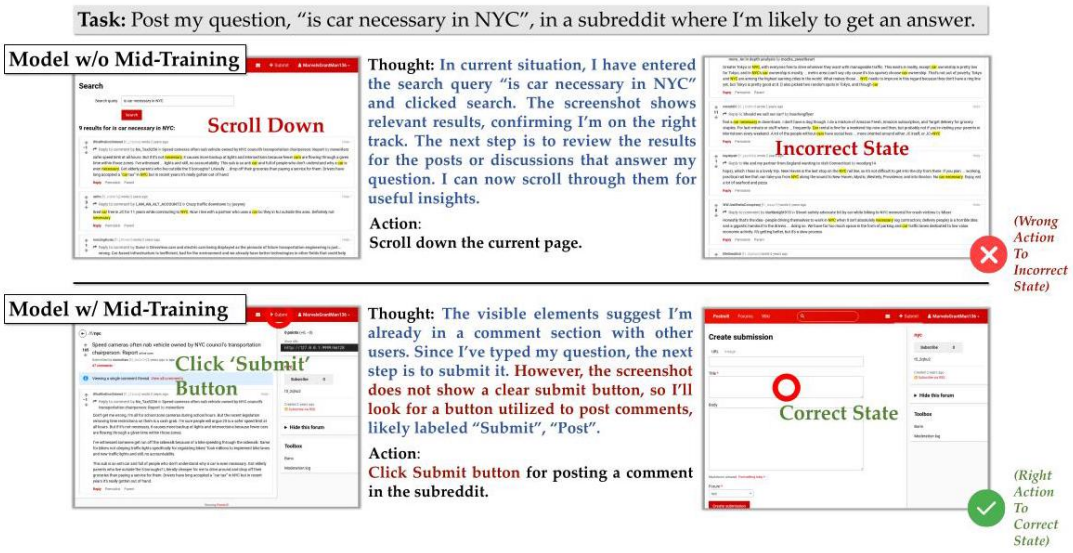

Figure 2: A case illustrating the performance of the Model w/o Mid-Training and the Model w/ Mid-Training under the same task. The middle text shows the model's thought process and the action taken, while the screenshots on the left and right represent the screen states before and after the action, respectively. The model with middle training (bottom) successfully reflects on errors and generates correct actions from error states, while the model without mid-training (top) fails to recover from such states.

图 2: 展示无中期训练模型与有中期训练模型在同一任务下的表现案例。中间文本显示模型的思考过程及采取的动作，左右截图分别代表动作前后的屏幕状态。带中期训练的模型 (下方) 能成功反思错误并从错误状态生成正确动作，而无中期训练模型 (上方) 则无法从此类状态恢复。

4.2 Evaluation Environment

4.2 评估环境

We use AndroidWorld (Rawles et al., 2024) and WebArena (Zhou et al., 2023) as our testbeds, as their dynamic nature makes them ideal environments for study the effects of different domains during the mid-training stage. For WebArena, we use the version processed in AgentBoard (Ma et al., 2024). We opt for interactive benchmarks over static benchmarks (Deng et al., 2023; Li et al., 2024) based on two considerations: (1) Static environments inherently contain annotation bias, wherein multiple valid paths exist to complete a task, yet annotator preferences result in only one path being labeled as correct for each step (e.g., finding a playlist through either search functionality or category navigation); (2) Our mid-training approach primarily enhances fundamental capabilities such as reasoning and perception rather than in-domain knowledge-for example, improving the model's ability to complete tasks in new websites through exploration and reflection. In contrast, step-by-step optimal action annotations rely heavily on the model's prior knowledge about specific websites or applications. We provide more evaluation details are provided in Appendix E.

我们使用 AndroidWorld(Rawles 等，2024) 和 WebArena(Zhou 等，2023) 作为测试平台，由于它们的动态特性，使其成为研究中期训练阶段不同领域影响的理想环境。对于 WebArena，我们采用了 AgentBoard(Ma 等，2024) 处理过的版本。我们基于两个考虑选择交互式基准测试而非静态基准测试 (Deng 等，2023；Li 等，2024):(1) 静态环境本质上存在标注偏差，任务存在多条有效路径，但标注者偏好导致每一步仅标注一条路径为正确 (例如，通过搜索功能或类别导航找到播放列表)；(2) 我们的中期训练方法主要提升推理和感知等基础能力，而非领域内知识——例如，通过探索和反思提升模型在新网站完成任务的能力。相比之下，逐步最优动作标注高度依赖模型对特定网站或应用的先验知识。更多评估细节见附录 E。

4.3 Results on Separate Domain

4.3 不同领域上的结果

We report the performances of different domains as mid-training on Webarena and Android-World (Table 3). Our analysis reveals several important findings:

我们报告了不同领域作为中期训练在 WebArena 和 AndroidWorld 上的表现 (表 3)。我们的分析揭示了若干重要发现:

Mathematical data generally improved the most: Both language-only and vision-language mathematical reasoning tasks demonstrate substantial performance improvements across benchmarks. In the language modality, models mid-trained with MathInstruct achieve the highest success rate on WebArena (10.9%) and strong performance on An-droidWorld (14.4%). Similarly, in the vision-language domain, Multi-modal Math shows impressive gains of 8.5% on WebArena and 15.3% on AndroidWorld. This consistent pattern suggests that mathematical reasoning capabilities, regardless of input modality, improve generalizable reasoning skills that transfer effectively to GUI agent tasks. A case study of a model mid-trained by MathInstruct (Yue et al., 2023) is shown in Figure 2. The task asks the agent to post the question ”Is car necessary in NYC” on the Reddit website. Both agents initially navigated to incorrect pages. However, the MathInstruct-trained agent

数学数据通常提升最大: 语言单模态和视觉-语言数学推理任务在各基准上均表现出显著性能提升。在语言模态中，使用 MathInstruct 中期训练的模型在 WebArena 上取得最高成功率 (10.9%)，在 AndroidWorld 上表现也很强 (14.4%)。同样，在视觉-语言领域，多模态数学任务在 WebArena 上提升 8.5%, 在 AndroidWorld 上提升 15.3%。这一持续的模式表明, 无论输入模态如何, 数学推理能力均能提升通用推理技能，有效迁移至图形用户界面 (GUI) 代理任务。图 2 展示了一个由 MathInstruct(Yue 等，2023) 中期训练的模型案例研究。该任务要求代理在 Reddit 网站发布问题“纽约市是否需要汽车”。两个代理最初均导航至错误页面。然而，经过 MathInstruct 训练的代理

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

领域	观察	WebArena		AndroidWorld
		PR	SR	SR
仅限 GUI 后训练	图像	26.3	6.2	9.0
公开基线				
GPT-4o-2024-11-20	图像	36.9	15.6	11.7
OS-Genesis-7B	图像 + 辅助功能树	-	-	17.4
AGUVIS-72B	图像	-	-	26.1
Claude3-Haiku	辅助功能树	26.8	12.7	-
Llama3-70b	辅助功能树	35.6	12.6	-
Gemini1.5-Flash	辅助功能树	32.4	11.1	-
视觉与语言模态				
图表/文档问答	图像	24.6	6.2	15.3
非 GUI 感知	图像	28.7	7.6	14.0
GUI 感知	图像	27.4	7.1	14.0
网页截图转代码	图像	28.0	6.6	9.9
非 GUI 代理	图像	30.8	8.5	13.5
多模态数学 ✓	图像	30.4	8.5	15.3
多轮视觉对话	图像	30.0	9.0	12.6
语言模态				
数学指导 ✓	图像	31.9	10.9	14.4
奥林匹克数学 ✓	图像	31.5	8.5	13.1
代码输入输出 ✓	图像	29.2	9.0	14.9
网页知识库	图像	31.3	9.5	9.0
领域组合 (采样自 ✓ 个领域的数据)				
GUIMid	图像	34.3	9.5	21.2

Table 3: Progress Rate (PR) and Success Rate (SR) of Qwen2-VL-7B-Instruct across various domains using a two-stage training strategy. Color-coded cells (green/red) are employed to denote improvements or declines relative to the post-training only baseline, with deeper shades indicating larger score shifts.

表 3:Qwen2-VL-7B-Instruct 在采用两阶段训练策略下，不同领域的进展率 (PR) 和成功率 (SR)。使用颜色编码单元 (绿色/红色) 表示相较于仅后期训练基线的提升或下降，颜色越深表示分数变化越大。

demonstrated better reasoning by thinking, ”However, the screenshot does not show a clear submit button, so I’ll look for a button utilized to post comments,” and subsequently located the correct ”submit” button. In contrast, the baseline model without mid-training became stuck on the incorrect page and continued scrolling up and down fruitlessly.

通过思考展现了更好的推理能力，“然而，截图中没有明显的提交按钮，所以我会寻找用于发布评论的按钮”，随后找到了正确的“提交”按钮。相比之下，未经过中期训练的基线模型卡在了错误的页面上，徒劳地上下滚动。

Strong cross-modal and cross-domain transfer: Language-only tasks demonstrate remarkable effectiveness for multi-modal GUI tasks. Olympiad Math achieves 31.5% progress and 8.5% success on WebArena, outperforming most vision-language tasks. Similarly, CodeI/O reaches a 14.9% success rate on AndroidWorld. Web Knowledge

Base shows effectiveness primarily in the web domain, likely due to its focus on web-specific information rather than mobile. These results suggest that conduct mid-training on text-only mathematics, code, and knowledge data can enhance fundamental abilities for GUI agents, even for multi-modal tasks-offering valuable insights for addressing the challenge of limited GUI in-domain training data.

强大的跨模态和跨领域迁移能力: 仅语言任务对多模态 GUI 任务表现出显著效果。奥林匹克数学 (Olympiad Math) 在 WebArena 上取得 31.5% 的进展率和 8.5% 的成功率，优于大多数视觉语言任务。同样，CodeI/O 在 AndroidWorld 上达到 14.9% 的成功率。Web 知识库主要在网页领域表现有效，可能因为其专注于网页特定信息而非移动端。这些结果表明，在仅文本的数学、代码和知识数据上进行中期训练，可以提升 GUI 代理的基础能力，即使是多模态任务——为解决 GUI 领域训练数据有限的问题提供了宝贵见解。

Non-GUI Agents data, despite its relatively small size (50K samples), demonstrates strong performance (30.8% progress, 8.5% success on WebArena; 13.5% success on AndroidWorld). This suggests that agent interaction patterns can transfer to some extent. Multi-round Visual Conversation yields balanced improvements across benchmarks (9.0% on WebArena, 12.6% on AndroidWorld). Chart/Document QA performs good on AndroidWorld(15.3%) but bad on the web setting, suggesting the different requirement of digital platforms. However, the Web Screenshot2Code and GUI Perception data does not help a lot, this may due to Qwen2-VL-7B-Instruct already be trained on GUI perception data.

非 GUI 代理数据，尽管规模较小 (5 万样本)，仍表现出强劲性能 (WebArena 上进展率 30.8%，成功率 8.5%；AndroidWorld 上成功率 13.5%)。这表明代理交互模式在一定程度上可以迁移。多轮视觉对话在各基准上均衡提升 (WebArena 9.0%，AndroidWorld 12.6%)。图表/文档问答在 AndroidWorld 表现良好 (15.3%)，但在网页环境中表现较差，反映出数字平台的不同需求。然而，网页截图转代码 (Web Screenshot2Code) 和 GUI 感知数据帮助不大，可能是因为 Qwen2-VL-7B-Instruct 已在 GUI 感知数据上训练过。

Results on Domain Combination Our motivations is to leverage larger quantities of non-GUI datasets to enhance VLMs' foundational capabilities, thereby enabling more effective learning from limited GUI trajectory data. To validate this, we combine the top-performing domains to construct a combined mid-training dataset: GUIMid, which consists of randomly sampled data (150K samples from MathInstruct, 20K from CodeI/O, 50K from Olympiads Math, and 80K from Multi-modal Math). We explore the scaling law of GUIMid. Specifically, for the 300K sample mid-training dataset, we maintain the same ratio of non-GUI to GUI data as in our 150K sample experiments to ensure training stability. Instead of

领域组合结果我们的动机是利用更多非 GUI 数据集来增强视觉语言模型 (VLMs) 的基础能力，从而更有效地从有限的 GUI 轨迹数据中学习。为验证这一点，我们将表现最好的领域数据合并，构建了一个组合中期训练数据集 GUIMid: 随机采样数据包括 15 万来自 MathInstruct，2 万来自 CodeI/O，5 万来自奥林匹克数学，8 万来自多模态数学。我们探索了 GUIMid 的规模效应。具体来说，对于 30 万样本的中期训练数据集，我们保持非 GUI 与 GUI 数据的比例与 15 万样本实验相同，以确保训练稳定性。取代

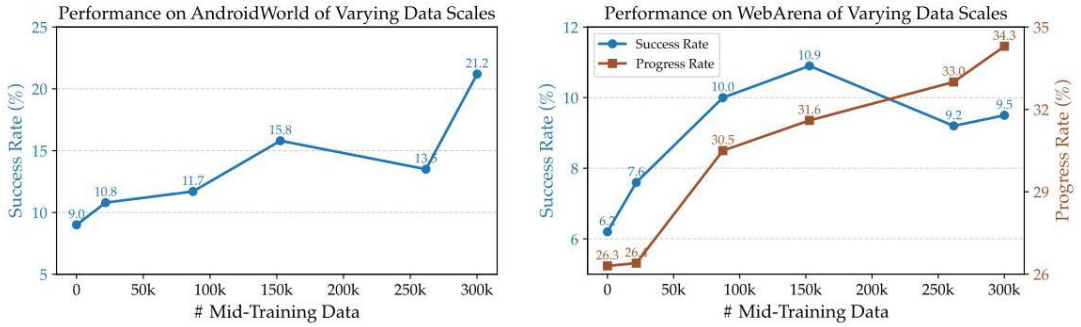

Figure 3: Performance of models trained on GUIMid with different scales.

图 3: 不同规模 GUIMid 训练模型的性能表现。

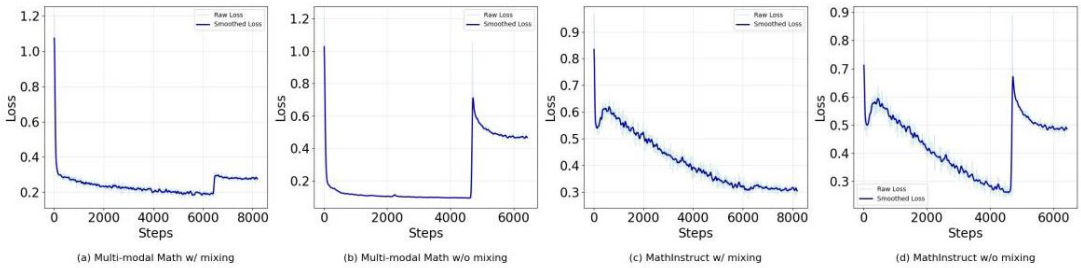

Figure 4: Comparison of training loss between two training strategies: (a) and (c) show the mixture of GUI trajectory data during mid-training, while (b) and (d) are not.

图 4: 两种训练策略的训练损失对比:(a) 和 (c) 显示中期训练中混合了 GUI 轨迹数据，而 (b) 和 (d) 则没有。

Domains	WebArena		AndroidWorld
	PR	SR	SR
MathInstruct (no mixing)	24.0	9.0	9.0
MathInstruct (mixing)	33.6	8.5	14.4
Multi-modal Math (no mixing)	25.4	6.2	14.9
Multi-modal Math (mixing)	30.4	8.5	15.3

领域	网络竞技场		安卓世界
	公关	系统请求	系统请求
数学教学 (不混合)	24.0	9.0	9.0
数学教学 (混合)	33.6	8.5	14.4
多模态数学 (不混合)	25.4	6.2	14.9
多模态数学 (混合)	30.4	8.5	15.3

Table 4: Progress Rate (PR) and Success Rate (SR) with and without GUI trajectory data integration during the mid-training stage. ”mixing” indicates the mid-training data is mixed with GUI trajectory data, while ”no mixing” indicates it was not.

表 4: 中期训练阶段有无集成 GUI 轨迹数据时的进展率 (PR) 和成功率 (SR)。“混合”表示中期训练数据与 GUI 轨迹数据混合，“不混合”表示未混合。

introducing new GUI data, we duplicate the existing GUI trajectory data. In Figure 3, the x-axis represents the effective mid-training data volume, calculated as the total training data volume multiplied by the fixed proportion allocated to mid-training samples. The model exhibits scaling laws with increasing mid-training data volume on both AndroidWorld and WebArena. While we observe a slight decrease in WebArena success rates around the 300K sample mark, the progress rate metric provides a more nuanced assessment of performance improvements. This metric, which captures fine-grained capability development, shows consistent growth: from approximately 26.4% at 21.8K samples to 31.6% at 152K samples, and further to 34.3% at 300K samples. This steady upward trajectory indicates the effective of the scaling law of GUIMid.

在引入新的 GUI 数据时, 我们复制了现有的 GUI 轨迹数据。图 3 中, x 轴表示有效的中期训练数据量, 计算方法为总训练数据量乘以分配给中期训练样本的固定比例。模型在 AndroidWorld 和 WebArena 上均表现出随着中期训练数据量增加的规模定律。虽然在 WebArena 上大约 300K 样本处成功率略有下降，但进展率指标提供了对性能提升更细致的评估。该指标捕捉了细粒度的能力发展，表现为持续增长: 从约 21.8K 样本时的 26.4% 增长到 152K 样本时的 31.6%，再到 300K 样本时的 34.3%。这一稳定上升趋势表明 GUIMid 的规模定律有效。

4.4 Ablation Study

4.4 消融研究

The effects of adding GUI trajectory data to the mid-training stage. We compare the loss curves (Figure 4) and performances (Table 4) when mixing/no mixing GUI trajectory data into the mid-training stage. Due to the domain gap between mid-training data and GUI trajectory data, domain switching may cause sharp fluctuations in the loss curve, leading

在中期训练阶段添加 GUI 轨迹数据的效果。我们比较了混合/不混合 GUI 轨迹数据进入中期训练阶段时的损失曲线 (图 4) 和性能表现 (表 4)。由于中期训练数据与 GUI 轨迹数据之间存在领域差异, 领域切换可能导致损失曲线出现剧烈波动，进而

Domains	Difficulty	WebArena		AndroidWorld
		PR	SR	SR
Orca-Math	Easy	31.9	10.0	9.9
Randomly Sampled Data	Middle	30.6	9.5	10.8
Olympiad Math	Hard	31.5	8.5	13.1

领域	难度	WebArena		AndroidWorld
		PR	SR	SR
Orca-Math	简单	31.9	10.0	9.9
随机抽样数据	中等	30.6	9.5	10.8
奥林匹克数学	困难	31.5	8.5	13.1

Table 5: The impact of mathematical difficulty in mid-training data. We sample three subsets from the NuminaMath dataset based on their difficulty levels.

表 5: 中期训练数据中数学难度的影响。我们根据难度等级从 NuminaMath 数据集中抽取了三个子集。

to instability (as shown in Figure 4, plots (b) and (d)), or even cause gradient overflow to NaN values. In contrast, plots (a) and (c) demonstrate significantly smoother training dynamics after merging the data. Table 4 reveals that after mixing GUI trajectory data, both MathInstruct and Multi-modal Math show significant performance improvements. In WebArena, MathInstruct's progress rate increased from 24.0 to 33.6 after mixing. Also, in AndroidWorld, Multi-modal Math's success rate improved from 14.9 to 15.3 after mixing.

导致不稳定 (如图 4 中图 (b) 和 (d) 所示)，甚至引起梯度溢出为 NaN 值。相比之下，图 (a) 和 (c) 显示合并数据后训练动态明显更平稳。表 4 显示混合 GUI 轨迹数据后，MathInstruct 和多模态数学模型的性能均有显著提升。在 WebArena 中，MathInstruct 的进步率从 24.0 提升至 33.6。同样，在 AndroidWorld 中，多模态数学模型的成功率从 14.9 提升至 15.3。

Performance with Respect to Mid-training Data Difficulty We analyze performance across three NuminaMath subsets of varying difficulty: the relatively easier Orca-Math (Mitra et al., 2024), the highly challenging Olympiad Math (LI et al., 2024), and a medium-difficulty random subset. Table 5 shows that performance in mobile environments generally improves with increased data difficulty, while WebArena exhibits no clear correlation. This difference likely stems from AndroidWorld's use of real applications with more diverse interaction patterns.

关于中期训练数据难度的性能分析我们分析了三个不同难度的 NuminaMath 子集的性能: 相对简单的 Orca-Math(Mitra 等，2024)、极具挑战性的奥林匹克数学 (LI 等，2024) 以及中等难度的随机子集。表 5 显示，在移动环境中，随着数据难度增加，性能普遍提升，而 WebArena 则未表现出明显相关性。这种差异可能源于 AndroidWorld 使用了具有更多样交互模式的真实应用。

5 Related Work

5 相关工作

GUI Agents. Recent advancements in GUI agents have led to diverse benchmark development. Non-interactive benchmarks for web (Mind2web (Deng et al., 2023), WebLINX (Lu et al., 2024b), WebVoyager (He et al., 2024)) and mobile environments (AITW (Rawles et al., 2023), AITZ (Zhang et al., 2024a), AndroidControl (Li et al., 2024)) have inherent limitations due to annotator bias when multiple valid task paths exist. To address these limitations, interactive benchmarks have emerged across mobile/desktop (OSWorld (Xie et al., 2024), AndroidWorld (Rawles et al., 2024)) and web environments (WebArena (Zhou et al., 2023), VisualWebArena (Koh et al., 2024a), WorkArena (Drouin et al., 2024), WebCanvas (Pan et al., 2024)). Our research on enhancing non-GUI capabilities (e.g., reasoning) requires evaluation frameworks where assessment is not bottlenecked by in-domain perception or knowledge limitations. VisualWebArena's trajectory-level success reporting may obscure reasoning improvements when GUI perception is the main bottleneck. While WebCanvas attempts to address this, most of its tasks are invalid because of the update of live websites. We therefore select AgentBoard's (Ma et al., 2024) subgoal-wise labeled versions of WebArena and AndroidWorld as our evaluation framework. This approach enables the isolation and measurement of specific capability shift despite potential limitations in other areas.

GUI 代理。近期 GUI 代理的进展推动了多样化基准的开发。针对网页 (Mind2web(Deng 等，2023)、WebLINX(Lu 等, 2024b)、WebVoyager(He 等, 2024)) 和移动环境 (AITW(Rawles 等, 2023)、AITZ(Zhang 等, 2024a)、AndroidControl(Li 等, 2024)) 的非交互式基准存在注释者偏差问题, 尤其当存在多条有效任务路径时。为解决这些限制, 交互式基准在移动/桌面 (OSWorld(Xie 等, 2024)、AndroidWorld(Rawles 等, 2024)) 和网页环境 (WebArena(Zhou 等, 2023)、VisualWebArena(Koh 等, 2024a)、WorkArena(Drouin 等，2024)、WebCanvas(Pan 等，2024)) 中出现。我们关于提升非 GUI 能力 (如推理) 的研究需要评估框架，避免因领域内感知或知识限制成为瓶颈。VisualWebArena 的轨迹级成功报告可能掩盖了当 GUI 感知为主要瓶颈时推理能力的提升。尽管 WebCanvas 尝试解决此问题，但其大部分任务因实时网站更新而失效。因此，我们选择 AgentBoard(Ma 等，2024) 对 WebArena 和 AndroidWorld 的子目标标注版本作为评估框架。该方法使得尽管其他方面存在潜在限制，仍能隔离并衡量特定能力的变化。

Mid-Training. Mid-training refers to the stage between pre-training and post-training designed to enhance foundational capabilities (Abdin et al., 2024), whereas post-training optimizes models to adapt to specific downstream tasks. Recent works like Phi 3.5 (Abdin et al., 2024), Yi-Lightning (Wake et al., 2024), OLMo 2 (OLMo et al., 2024) and CodeI/O (Li et al., 2025) have demonstrated the effectiveness of strategic mid-training interventions-use mid-training to enhance the foundational abilities like context length, multilingual knowledge and code understanding. For GUI agents, where high-quality trajectories are scarce, mid-training becomes particularly valuable. However, current research lacks systematic exploration of out-of-domain mid-training techniques specifically tailored for GUI agents-a critical gap that our work addresses.

中期训练。中期训练指介于预训练和后训练之间的阶段，旨在增强基础能力 (Abdin 等，2024)，而后训练则优化模型以适应特定下游任务。近期工作如 Phi 3.5(Abdin 等，2024)、Yi-Lightning(Wake 等，2024)、OLMo 2(OLMo 等，2024) 和 CodeI/O(Li 等，2025) 展示了战略性中期训练干预的有效性——利用中期训练提升上下文长度、多语言知识和代码理解等基础能力。对于高质量轨迹稀缺的 GUI 代理，中期训练尤为重要。然而，当前研究缺乏针对 GUI 代理的领域外中期训练技术的系统探索——这是我们工作的关键突破点。

6 Conclusion

6 结论

In this paper, we propose using mid-training to enhance GUI agents' foundational capabilities, enabling more effective learning from limited trajectory data. While GUI-specific

本文提出利用中期训练提升 GUI 代理的基础能力，使其能更有效地从有限轨迹数据中学习。尽管 GUI 特定

data remains scarce, there is much more non-GUI data such as mathematical reasoning, and coding. We explore 11 diverse non-GUI tasks, demonstrating for the first time the significant impact of mathematical reasoning data on GUI task performance, as well as the surprising effectiveness of text-only mathematical reasoning in improving multimodal GUI agent capabilities. Our findings also reveal how non-GUI tasks such as text web knowledge and embodied agent trajectories can substantially enhance GUI agent performance. These results provide valuable insights for the research community in constructing more effective GUI training pipelines. We will

open-source all data, models, and training recipes to facilitate future research in the GUI agent domain.

数据依然稀缺，但存在大量非 GUI 数据，如数学推理和编码。我们探索了 11 个多样的非 GUI 任务，首次展示了数学推理数据对 GUI 任务性能的显著影响，以及仅文本数学推理在提升多模态 GUI 代理能力上的惊人效果。我们的研究还揭示，诸如文本网页知识和具身代理轨迹等非 GUI 任务能显著增强 GUI 代理性能。这些结果为研究社区构建更有效的 GUI 训练流程提供了宝贵见解。我们将开源所有数据、模型和训练方案，以促进 GUI 代理领域的未来研究。

References

参考文献

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. arXiv preprint arXiv:2404.14219, 2024.

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, 等人。Phi-3 技术报告: 一款可在手机本地运行的高性能语言模型。arXiv 预印本 arXiv:2404.14219，2024 年。

Łukasz Borchmann, Michał Pietruszka, Tomasz Stanislawek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler, and Filip Graliński. Due: End-to-end document understanding benchmark. In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2), 2021.

Łukasz Borchmann, Michał Pietruszka, Tomasz Stanislawek, Dawid Jurkiewicz, Michał Turski, Karolina Szyndler 和 Filip Graliński。Due: 端到端文档理解基准。在第三十五届神经信息处理系统会议数据集与基准赛道 (第二轮)，2021 年。

Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo, et al. Guicourse: From general vision language models to versatile gui agents. arXiv preprint arXiv:2406.11317, 2024.

Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo 等。Guicourse: 从通用视觉语言模型到多功能 GUI 代理。arXiv 预印本 arXiv:2406.11317，2024 年。

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse 和 John Schulman。训练验证器以解决数学文字题。arXiv 预印本 arXiv:2110.14168，2021 年。

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. Advances in Neural Information Processing Systems, 36:28091-28114,

2023.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun 和 Yu Su。Mind2web: 迈向通用网络代理。神经信息处理系统进展，36:28091-28114，2023 年。

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, et al. Workarena: How capable are web agents at solving common knowledge work tasks? arXiv preprint arXiv:2403.07718, 2024.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez 等。Workarena: 网络代理在解决常识性工作任务中的能力如何？arXiv 预印本 arXiv:2403.07718，2024 年。

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. Navigating the digital world as humans do: Universal visual grounding for GUI agents. In The Thirteenth International Conference on Learning Representations, 2025. URL https://openreview.net/forum?id=kxnoqaisCT.

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun 和 Yu Su。像人类一样导航数字世界: 面向 GUI 代理的通用视觉定位。在第十三届国际学习表征会议，2025 年。网址 https://openreview.net/forum?id=kxnoqaisCT。

Jarvis Guo, Tuney Zheng, Yuelin Bai, Bo Li, Yubo Wang, King Zhu, Yizhi Li, Graham Neubig, Wenhu Chen, and Xiang Yue. Mammoth-vl: Eliciting multimodal reasoning with instruction tuning at scale. 2024. URL https://arxiv.org/abs/2412.05237.

Jarvis Guo, Tuney Zheng, Yuelin Bai, Bo Li, Yubo Wang, King Zhu, Yizhi Li, Graham Neubig, Wenhu Chen 和 Xiang Yue。Mammoth-vl: 通过大规模指令调优激发多模态推理。2024 年。网址 https://arxiv.org/abs/2412.05237。

Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. A real-world webagent with planning, long context understanding, and program synthesis. arXiv preprint arXiv:2307.12856, 2023.

Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck 和 Aleksandra Faust。具备规划、长上下文理解和程序合成能力的真实世界网络代理。arXiv 预印本 arXiv:2307.12856，2023 年。

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 6864-6890, 2024.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan 和 Dong Yu。Webvoyager: 基于大型多模态模型构建端到端网络代理。在第 62 届计算语言学协会年会论文集 (第一卷: 长文)，页 6864-6890，2024 年。

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-40 system card. arXiv preprint arXiv:2410.21276, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford 等。Gpt-40 系统卡。arXiv 预印本 arXiv:2410.21276，2024 年。

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. Dvqa: Understanding data visualizations via question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5648-5656, 2018.

Kushal Kafle, Brian Price, Scott Cohen 和 Christopher Kanan。DVQA: 通过问答理解数据可视化。在 IEEE 计算机视觉与模式识别会议论文集，页 5648-5656，2018 年。

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 881-905, 2024a.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov 和 Daniel Fried。Visualwebarena: 评估多模态代理在真实视觉网络任务中的表现。在第 62 届计算语言学协会年会论文集 (第一卷: 长文)，页 881-905，2024a 年。

Jing Yu Koh, Stephen McAleer, Daniel Fried, and Ruslan Salakhutdinov. Tree search for language model agents. arXiv preprint arXiv:2407.01476, 2024b.

Jing Yu Koh, Stephen McAleer, Daniel Fried 和 Ruslan Salakhutdinov。语言模型代理的树搜索。arXiv 预印本 arXiv:2407.01476，2024b 年。

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Nu-minamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/ project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf), 2024.

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, 和 Stanislas Polu. Nu-minamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf), 2024。

Junlong Li, Daya Guo, Dejian Yang, Runxin Xu, Yu Wu, and Junxian He. Codei/o: Condensing reasoning patterns via code input-output prediction. arXiv preprint arXiv:2502.07316, 2025.

Junlong Li, Daya Guo, Dejian Yang, Runxin Xu, Yu Wu, 和 Junxian He. Codei/o: 通过代码输入输出预测凝练推理模式。arXiv 预印本 arXiv:2502.07316, 2025。

Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on ui control agents. Advances in Neural Information Processing Systems, 37:92130-92154, 2024.

Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, 和 Oriana Riva. 数据规模对用户界面控制代理影响的研究。神经信息处理系统进展, 37:92130-92154, 2024。

Junpeng Liu, Tianyue Ou, Yifan Song, Yuxiao Qu, Wai Lam, Chenyan Xiong, Wenhu Chen, Graham Neubig, and Xiang Yue. Harnessing webpage uis for text-rich visual understanding. arXiv preprint arXiv:2410.13824, 2024a.

Junpeng Liu, Tianyue Ou, Yifan Song, Yuxiao Qu, Wai Lam, Chenyan Xiong, Wenhu Chen, Graham Neubig, 和 Xiang Yue. 利用网页用户界面进行文本丰富的视觉理解。arXiv 预印本 arXiv:2410.13824, 2024a。

Wei Liu, Junlong Li, Xiwen Zhang, Fan Zhou, Yu Cheng, and Junxian He. Diving into self-evolving training for multimodal reasoning. arXiv preprint arXiv:2412.17451, 2024b.

Wei Liu, Junlong Li, Xiwen Zhang, Fan Zhou, Yu Cheng, 和 Junxian He. 深入自我进化训练以实现多模态推理。arXiv 预印本 arXiv:2412.17451, 2024b。

Quanfeng Lu, Wenqi Shao, Zitao Liu, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, Yu Qiao, and Ping Luo. Gui odyssey: A comprehensive dataset for cross-app gui navigation on mobile devices. arXiv preprint arXiv:2406.08451, 2024a.

Quanfeng Lu, Wenqi Shao, Zitao Liu, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, Yu Qiao, 和 Ping Luo. Gui Odyssey: 面向移动设备跨应用 GUI 导航的综合数据集。arXiv 预印本 arXiv:2406.08451, 2024a。

Xing Han Lu, Zdeněk Kasner, and Siva Reddy. Weblinx: Real-world website navigation with multi-turn dialogue. In International Conference on Machine Learning, pp. 33007-33056. PMLR, 2024b.

Xing Han Lu, Zdeněk Kasner, 和 Siva Reddy. Weblinx: 基于多轮对话的真实网站导航。国际机器学习会议论文集，页码 33007-33056。PMLR, 2024b。

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. Agentboard: An analytical evaluation board of multi-turn llm agents. Advances in Neural Information Processing Systems, 37:74325-74362, 2024.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, 和 Junxian He. Agentboard: 多轮大型语言模型代理的分析评估平台。神经信息处理系统进展, 37:74325-74362, 2024。

Nitesh Methani, Pritha Ganguly, Mitesh M Khapra, and Pratyush Kumar. Plotqa: Reasoning over scientific plots. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1527-1536, 2020.

Nitesh Methani, Pritha Ganguly, Mitesh M Khapra, 和 Pratyush Kumar. Plotqa: 科学图表上的推理。IEEE/CVF 冬季计算机视觉应用会议论文集，页码 1527-1536, 2020。

Arindam Mitra, Hamed Khanpour, Corby Rosset, and Ahmed Awadallah. Orca-math: Unlocking the potential of slms in grade school math. arXiv preprint arXiv:2402.14830, 2024.

Arindam Mitra, Hamed Khanpour, Corby Rosset, 和 Ahmed Awadallah. Orca-math: 释放大规模语言模型 (SLMs) 在小学数学中的潜力。arXiv 预印本 arXiv:2402.14830, 2024。

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 olmo 2 furious. arXiv preprint arXiv:2501.00656, 2024.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, 等。2 olmo 2 furious。arXiv 预印本 arXiv:2501.00656, 2024。

OpenAI. Computer-Using Agent. https://openai.com/index/computer-using-agent/, January 2025.

OpenAI. 计算机使用代理。https://openai.com/index/computer-using-agent/, 2025 年 1 月。

Tianyue Ou, Frank F. Xu, Aman Madaan, Jiarui Liu, Robert Lo, Abishek Sridhar, Sudipta Sengupta, Dan Roth, Graham Neubig, and Shuyan Zhou. Synatra: Turning indirect knowledge into direct demonstrations for digital agents at scale. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024. URL https: //openreview.net/forum?id=KjNEzWRIqn.

Tianyue Ou, Frank F. Xu, Aman Madaan, Jiarui Liu, Robert Lo, Abishek Sridhar, Sudipta Sengupta, Dan Roth, Graham Neubig, 和 Shuyan Zhou. Synatra: 将间接知识转化为大规模数字代理的直接示范。第三十八届神经信息处理系统年会论文集，2024。网址 https://openreview.net/forum?id=KjNEzWRIqn。

Yichen Pan, Dehan Kong, Sida Zhou, Cheng Cui, Yifei Leng, Bing Jiang, Hangyu Liu, Yanyi Shang, Shuyan Zhou, Tongshuang Wu, et al. Webcanvas: Benchmarking web agents in online environments. arXiv preprint arXiv:2406.12373, 2024.

Yichen Pan, Dehan Kong, Sida Zhou, Cheng Cui, Yifei Leng, Bing Jiang, Hangyu Liu, Yanyi Shang, Shuyan Zhou, Tongshuang Wu, 等。Webcanvas: 在线环境中网页代理的基准测试。arXiv 预印本 arXiv:2406.12373, 2024。

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. An-droidinthewild: A large-scale dataset for android device control. Advances in Neural Information Processing Systems, 36:59708-59728, 2023.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, 和 Timothy Lillicrap. Androidinthewild: 用于安卓设备控制的大规模数据集。神经信息处理系统进展，36:59708-59728, 2023。

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, et al. Android-world: A dynamic benchmarking environment for autonomous agents. arXiv preprint arXiv:2405.14573, 2024.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala 等. Android-world: 用于自主代理的动态基准测试环境。arXiv 预印本 arXiv:2405.14573, 2024。

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See Kiong Ng, Lidong Bing, and Roy Lee. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. In Findings of the Association for Computational Linguistics: EMNLP 2024, pp. 4663-4680, 2024.

Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See Kiong Ng, Lidong Bing, 和 Roy Lee. Math-llava: 为多模态大型语言模型引导数学推理。发表于计算语言学协会会议论文集:EMNLP 2024, 页 4663-4680, 2024。

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning Text and Embodied Environments for Interactive Learning. In Proceedings of the International Conference on Learning Representations (ICLR), 2021. URL https://arxiv.org/abs/2010.03768.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, 和 Matthew Hausknecht. ALFWorld: 对齐文本与具身环境以实现交互式学习。发表于国际学习表征会议 (ICLR) 论文集，2021。网址 https://arxiv.org/abs/2010.03768。

Yueqi Song, Frank F Xu, Shuyan Zhou, and Graham Neubig. Beyond browsing: Api-based web agents. 2024.

Yueqi Song, Frank F Xu, Shuyan Zhou, 和 Graham Neubig. 超越浏览: 基于 API 的网页代理。2024。

Qiushi Sun, Zhirui Chen, Fangzhi Xu, Kanzhi Cheng, Chang Ma, Zhangyue Yin, Jian-ing Wang, Chengcheng Han, Renyu Zhu, Shuai Yuan, et al. A survey of neural code intelligence: Paradigms, advances and beyond. arXiv preprint arXiv:2403.14734, 2024a.

Qiushi Sun, Zhirui Chen, Fangzhi Xu, Kanzhi Cheng, Chang Ma, Zhangyue Yin, Jian-ing Wang, Chengcheng Han, Renyu Zhu, Shuai Yuan 等. 神经代码智能综述: 范式、进展及未来。arXiv 预印本 arXiv:2403.14734, 2024a。

Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu, et al. Os-genesis: Automating gui agent trajectory construction via reverse task synthesis. arXiv preprint arXiv:2412.19723, 2024b.

Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu 等. Os-genesis: 通过逆向任务合成自动构建 GUI 代理轨迹。arXiv 预印本 arXiv:2412.19723, 2024b。

Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. Hierarchical multimodal transformers for multipage docvqa. Pattern Recognition, 144:109834, 2023.

Rubèn Tito, Dimosthenis Karatzas, 和 Ernest Valveny. 用于多页文档视觉问答的分层多模态变换器。模式识别，144:109834, 2023。

Alan Wake, Bei Chen, CX Lv, Chao Li, Chengen Huang, Chenglin Cai, Chujie Zheng, Daniel Cooper, Fan Zhou, Feng Hu, et al. Yi-lightning technical report. arXiv preprint arXiv:2412.01253, 2024.

Alan Wake, Bei Chen, CX Lv, Chao Li, Chengen Huang, Chenglin Cai, Chujie Zheng, Daniel Cooper, Fan Zhou, Feng Hu 等. Yi-lightning 技术报告。arXiv 预印本 arXiv:2412.01253, 2024。

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qw en2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge 等. Qw en2-vl: 提升视觉语言模型对任意分辨率世界的感知能力。arXiv 预印本 arXiv:2409.12191, 2024a。

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better llm agents. In Forty-first International Conference on Machine Learning, 2024b.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, 和 Heng Ji. 可执行代码操作激发更优的大型语言模型代理。发表于第四十一届国际机器学习大会，2024b。

Zhiyong Wu, Chengcheng Han, Zichen Ding, Zhenmin Weng, Zhoumianze Liu, Shunyu Yao, Tao Yu, and Lingpeng Kong. Os-copilot: Towards generalist computer agents with self-improvement. arXiv preprint arXiv:2402.07456, 2024.

Zhiyong Wu, Chengcheng Han, Zichen Ding, Zhenmin Weng, Zhoumianze Liu, Shunyu Yao, Tao Yu, 和 Lingpeng Kong. Os-copilot: 迈向具备自我提升能力的通用计算机代理。arXiv 预印本 arXiv:2402.07456, 2024。

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Jing Hua Toh, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. Advances in Neural Information Processing Systems, 37:52040-52094, 2024.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Jing Hua Toh, Zhoujun Cheng, Dongchan Shin, Fangyu Lei 等. Osworld: 在真实计算机环境中评测多模态代理的开放式任务能力。神经信息处理系统进展，37:52040-52094, 2024。

Yiheng Xu, Dunjie Lu, Zhennan Shen, Junli Wang, Zekun Wang, Yuchen Mao, Caiming Xiong, and Tao Yu. Agenttrek: Agent trajectory synthesis via guiding replay with web tutorials. arXiv preprint arXiv:2412.09605, 2024a.

Yiheng Xu, Dunjie Lu, Zhennan Shen, Junli Wang, Zekun Wang, Yuchen Mao, Caiming Xiong, 和 Tao Yu. Agenttrek: 通过网络教程引导回放合成代理轨迹。arXiv 预印本 arXiv:2412.09605, 2024a。

Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lu, Tianbao Xie, Amrita Saha, Doyen Sahoo, Tao Yu, and Caiming Xiong. Aguvis: Unified pure vision agents for autonomous gui interaction, 2024b.

徐一恒，王泽坤，王俊利，卢敦杰，谢天宝，Amrita Saha，Doyen Sahoo，余涛，熊才明。Aguvis: 用于自主 GUI 交互的统一纯视觉代理，2024b。

Jianwei Yang, Reuben Tan, Qianhui Wu, Ruijie Zheng, Baolin Peng, Yongyuan Liang, Yu Gu, Mu Cai, Seonghyeon Ye, Joel Jang, et al. Magma: A foundation model for multimodal ai agents. arXiv preprint arXiv:2502.13130, 2025.

杨建伟，Reuben Tan，吴千惠，郑瑞杰，彭宝林，梁永远，顾宇，蔡牧，Ye Seonghyeon，Joel Jang 等。Magma: 多模态 AI 代理的基础模型。arXiv 预印本 arXiv:2502.13130，2025。

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, et al. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. arXiv preprint arXiv:2310.05126, 2023.

叶嘉博，胡安文，徐海洋，叶庆浩，闫明，徐国海，李晨亮，田俊峰，钱琦，张骥等。Ureader: 基于多模态大语言模型的通用无 OCR 视觉语境语言理解。arXiv 预印本 arXiv:2310.05126，2023。

Xiao Yu, Baolin Peng, Vineeth Vajipey, Hao Cheng, Michel Galley, Jianfeng Gao, and Zhou Yu. Exact: Teaching ai agents to explore with reflective-mcts and exploratory learning. arXiv preprint arXiv:2410.02052, 2024.

余晓，彭宝林，Vineeth Vajipey，程浩，Michel Galley，高建峰，余舟。Exact: 通过反思性蒙特卡洛树搜索 (reflective-MCTS) 和探索性学习教导 AI 代理探索。arXiv 预印本 arXiv:2410.02052，2024。

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mammoth: Building math generalist models through hybrid instruction tuning. arXiv preprint arXiv:2309.05653, 2023.

岳翔，曲兴伟，张戈，付尧，黄文浩，孙焕，苏宇，陈文虎。Mammoth: 通过混合指令调优构建数学通用模型。arXiv 预印本 arXiv:2309.05653，2023。

Sukmin Yun, Haokun Lin, Rusiru Thushara, Mohammad Qazim Bhat, Yongxin Wang, Zutao Jiang, Mingkai Deng, Jinhong Wang, Tianhua Tao, Junbo Li, et al. Web2code: A large-scale webpage-to-code dataset and evaluation framework for multimodal llms. arXiv preprint arXiv:2406.20098, 2024.

尹淑敏，林浩坤，Rusiru Thushara，Mohammad Qazim Bhat，王永新，姜祖涛，邓明凯，王金宏，陶天华，李军波等。Web2code: 大规模网页到代码数据集及多模态大语言模型评估框架。arXiv 预印本 arXiv:2406.20098，2024。

Jiwen Zhang, Jihao Wu, Teng Yihua, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. Android in the zoo: Chain-of-action-thought for gui agents. In Findings of the Association for Computational Linguistics: EMNLP 2024, pp. 12016-12031, 2024a.

张继文，吴继豪，滕一华，廖明辉，徐诺，肖晓，魏中宇，唐杜宇。Android in the zoo: 面向 GUI 代理的行动思维链。发表于计算语言学协会会议成果:EMNLP 2024，页 12016-12031，2024a。

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Yichi Zhang, Ziyu Guo, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, et al. Mavis: Mathematical visual instruction tuning. arXiv e-prints, pp. arXiv-2407, 2024b.

张仁睿，魏欣宇，姜东志，张奕驰，郭子瑜，童成卓，刘嘉明，周奥军，魏斌，张尚航等。Mavis: 数学视觉指令调优。arXiv 电子预印本，页 arXiv-2407，2024b。

Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning. arXiv preprint arXiv:2307.04087, 2023.

赵博, 吴博雅, 何牧阳, 黄铁军。Svit: 视觉指令调优的规模化。arXiv 预印本 arXiv:2307.04087，2023。

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v (ision) is a generalist web agent, if grounded. arXiv preprint arXiv:2401.01614, 2024a.

郑博远，苟博宇，Kil Jihyung，孙焕，苏宇。GPT-4V(ision) 是通用型网页代理，前提是有基础支撑。arXiv 预印本 arXiv:2401.01614，2024a。

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v(ision) is a generalist web agent, if grounded. In Forty-first International Conference on Machine Learning, 2024b. URL https://openreview.net/forum?id=piecKJ2DlB.

郑博远，苟博宇，Kil Jihyung，孙焕，苏宇。GPT-4V(ision) 是通用型网页代理，前提是有基础支撑。发表于第四十一届国际机器学习大会，2024b。网址 https://openreview.net/forum?id=piecKJ2DlB。

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. arXiv preprint arXiv:2307.13854, 2023.

周书妍, Frank F Xu, 朱浩, 周旭辉, Robert Lo, Abishek Sridhar, 程贤义, 欧天悦, Yonatan Bisk, Daniel Fried 等。Webarena: 用于构建自主代理的真实网页环境。arXiv 预印本 arXiv:2307.13854，2023。

Action	Description
click [[x] [y]]	Click at coordinates (x, y).
type [[x] [y]] [value]	Type content into a field by coordinate.
scroll [[x] [y]] [value]	Scroll the page or a specific element.
go_back	Navigate to the previous screen or page.
go_home	Navigate to the home screen.
long_press[[x] [y]]	Long press on an coordinate.
enter	Press the Enter key.
open_app [app_name]	Open an app by [app_name].
wait [value]	Wait for the screen to update for [value] seconds.
stop [answer]	Stop the task with a goal status or answer.

操作	描述
点击 [[x] [y]]	在坐标 (x, y) 处点击。
输入 [[x] [y]] [value]	在指定坐标的输入框中输入内容。
滚动 [[x] [y]] [value]	滚动页面或特定元素。
返回	导航到上一个屏幕或页面。
回到首页	导航到主屏幕。
长按 [[x] [y]]	在坐标处长按。
回车	按下回车键。
打开应用 [app_name]	通过 [app_name] 打开应用。
等待 [value]	等待屏幕更新 [value] 秒。
停止 [answer]	以目标状态或答案停止任务。

Table 6: The action space for mobile tasks.

表 6: 移动任务的动作空间。

Action	Description
click [[x] [y]]	Click at coordinates (x, y).
type [[x] [y]] [value]	Type content into a field by coordinate.
clear [[x] [y]]	Clear the content of an element.
hover [[x] [y]]	Hover over an element by coordinate.
press [keys]	Press a key combination (e.g., Ctrl+v).
scroll [value]	Scroll the page.
new_tab	Open a new tab.
page_focus [tab_index]	Switch to a specific tab.
close_tab	Close the current tab.
goto [url]	Navigate to a URL.
go_back	Go to the previous page.
go_forward	Go to the next page.
stop [answer]	Issue this action when the task is considered complete.

操作	描述
点击 [[x] [y]]	在坐标 (x, y) 处点击。
输入 [[x] [y]] [value]	在指定坐标的字段中输入内容。
清除 [[x] [y]]	清除元素内容。
悬停 [[x] [y]]	在指定坐标悬停元素。
按下 [keys]	按下按键组合 (例如 Ctrl+v)。
滚动 [value]	滚动页面。
新标签页	打开新标签页。
页面聚焦 [tab_index]	切换到指定标签页。
关闭标签页	关闭当前标签页。
跳转到 [url]	导航到指定网址。
后退	返回上一页。
前进	前往下一页。
停止 [answer]	任务完成时执行此操作。

Table 7: The action space for web tasks.

表 7: 网页任务的动作空间。

A Details of the GUI Agent

A GUI 代理的详细信息

Action Space. During the data processing stage, we standardized the action spaces separately for mobile and web domains to achieve unified representations suitable for joint training. In the evaluation stage, we further aligned actions across different benchmarks by mapping them to the corresponding standardized action spaces. The actions involved in both the training and evaluation phases fall within the categories detailed in Table 6 for the mobile domain and Table 7 for the web domain.

动作空间。在数据处理阶段，我们分别对移动端和网页端的动作空间进行了标准化，以实现适合联合训练的统一表示。在评估阶段，我们通过将动作映射到相应的标准化动作空间，进一步对不同基准之间的动作进行了对齐。训练和评估阶段涉及的动作均属于移动端的表 6 和网页端的表 7 中详细列出的类别。

Observation space. To simulate real-world human interactions with GUI environments and to better investigate whether middle-training data enhances the reasoning ability of GUI agents, the observation space in this study consists solely of visual information. Specifically, during task execution, GUI agents receive only visual feedback from the screen and historical action information, without leveraging any additional textual information from the environment (e.g., DOM or accessibility trees). Such textual information could introduce extra information gain, potentially confounding the effect of middle-training data.

观测空间。为了模拟真实世界中人类与 GUI 环境的交互，并更好地研究中期训练数据是否提升了 GUI 代理的推理能力，本研究中的观测空间仅包含视觉信息。具体而言，在任务执行过程中，GUI 代理仅接收来自屏幕的视觉反馈和历史动作信息，不利用环境中的任何额外文本信息 (例如 DOM 或辅助功能树)。此类文本信息可能带来额外的信息增益，进而干扰中期训练数据的效果评估。

B Details of Mid-Training Data

B 中期训练数据的详细信息

Vision-language data generally comprises paired visual and textual information, such as image captions, annotated screenshots, and visually grounded instructions. Considering the inherently multi-modal nature of GUI Agents, leveraging vision-language data may facilitate better alignment between visual and textual modalities, potentially improving agents' comprehension of and interaction with graphical interfaces. Our primary focus for vision-language modalities includes:

视觉-语言数据通常包含配对的视觉和文本信息，如图像说明、带注释的截图和视觉基础指令。考虑到 GUI 代理的多模态本质，利用视觉-语言数据有助于实现视觉与文本模态的更好对齐，可能提升代理对图形界面的理解和交互能力。我们主要关注的视觉-语言模态包括:

(1) Chart/ Document Question Answering: Data in this category enhance the model's ability to perform reasoning-intensive tasks over visual representations of struc-

(1) 图表/文档问答: 该类别数据增强模型对结构化图表和文档视觉表示进行推理密集型任务的能力。我们的训练数据通过从 InfographicVQA 和 MAmmoTH-VL(Guo 等，2024) 中的 Ureader QA 随机抽取约 56K 样本，从 MPDocVQA(Tito 等，2023) 抽取 500 个样本，以及从 MathV360k(Liu 等，2024b) 的预热数据中抽取 93.5K 样本构建。总计生成约 15 万样本的中期训练数据集。

tured charts and documents. Our training data is constructed by randomly sampling approximately 56K samples from InfographicVQA and Ureader QA in MAmmoTH-VL (Guo et al., 2024), 500 samples from MPDocVQA (Tito et al., 2023), and 93.5K samples from the warm-up data of MathV360k (Liu et al., 2024b). In total, this process yields a dataset of 150K samples for mid-training.

tured charts and documents. Our training data is constructed by randomly sampling approximately 56K samples from InfographicVQA and Ureader QA in MAmmoTH-VL (Guo et al., 2024), 500 samples from MPDocVQA (Tito et al., 2023), and 93.5K samples from the warm-up data of MathV360k (Liu et al., 2024b). In total, this process yields a dataset of 150K samples for mid-training.

(2) Non-GUI Perception: This category enhances the model's perception capabilities for non-GUI images, such as posters, tables, and documents. Given the abundance of non-GUI perception datasets in online databases (e.g., documents (Kafle et al., 2018), posters (Ye et al., 2023), and plots (Methani et al., 2020)), we investigate whether leveraging such data can improve performance on GUI-related tasks. Specifically, we construct the training data by integrating 6.1K samples from Ureader OCR (Ye et al., 2023) with 143.9K randomly selected samples from DUE (Borchmann et al., 2021), yielding a total of 150k samples for the mid-training phase.

(2) 非 GUI 感知: 该类别增强模型对非 GUI 图像 (如海报、表格和文档) 的感知能力。鉴于在线数据库中存在大量非 GUI 感知数据集 (如文档 (Kafle 等，2018)、海报 (Ye 等，2023) 和图表 (Methani 等，2020))，我们探讨利用此类数据是否能提升 GUI 相关任务的表现。具体而言，我们通过整合 Ureader OCR(Ye 等，2023) 的 6.1K 样本与 DUE(Borchmann 等，2021) 中随机选取的 143.9K 样本构建训练数据，总计 15 万样本用于中期训练阶段。

(3) GUI Perception: This category aims to enhance the model's perceptual capabilities for GUI images. To achieve this, the training dataset is constructed by randomly sampling 50K instances from each of the Action Prediction, Webpage Question Answering, and Image Question Answering subsets within MultiUI (Liu et al., 2024a), resulting in a total of 150K samples for the mid-training stage.

(3) GUI 感知: 该类别旨在提升模型对 GUI 图像的感知能力。为此，训练数据集通过从 MultiUI(Liu 等，2024a) 中的动作预测、网页问答和图像问答子集各随机抽取 5 万实例构建，合计 15 万样本用于中期训练阶段。

(4) Multi-modal Math: Math data (Cobbe et al., 2021; Shi et al., 2024) has been widely used to enhance the reasoning capabilities of LLMs and VLLMs. We explore whether incorporating multi-modal mathematical data can further enhance the planning capabilities of GUI-Agent. Specifically, we construct the training dataset by randomly sampling 150K multi-modal math problems from the Mavis dataset (Zhang et al., 2024b), which comprises high-quality mathematical questions accompanied by comprehensive reasoning processes.

(4) 多模态数学: 数学数据 (Cobbe 等，2021；Shi 等，2024) 被广泛用于增强大型语言模型 (LLMs) 和视觉语言大型模型 (VLLMs) 的推理能力。我们探索引入多模态数学数据是否能进一步提升 GUI 代理的规划能力。具体而言，我们通过从 Mavis 数据集 (Zhang 等，2024b) 随机抽取 15 万多模态数学问题构建训练数据集，该数据集包含高质量的数学题目及其详尽的推理过程。

(5) Multi-round Visual Conversation: Agent trajectories often consist of multiple steps, requiring the model to understand and memorize previous steps to inform current decisions. Multi-round visual conversation data exhibits similar characteristics, as generating a response in a given turn typically depends on the context of prior turns. We examine whether multi-turn multi-modal question-answering data can enhance the performance of GUI-Agent tasks. Specifically, we construct the training dataset by randomly sampling 150 K multi-turn dialogues from the SVIT dataset (Zhao et al., 2023), which comprises multi-turn question-answering interactions involving intricate, image-based reasoning.

(5) 多轮视觉对话: 代理轨迹通常包含多个步骤，要求模型理解并记忆前序步骤以指导当前决策。多轮视觉对话数据具有类似特征，因为生成某一轮的回答通常依赖于前几轮的上下文。我们考察多轮多模态问答数据是否能提升 GUI 代理任务的表现。具体而言，我们通过从 SVIT 数据集 (Zhao 等，2023) 随机抽取 150 K 多轮对话构建训练数据集，该数据集包含涉及复杂图像推理的多轮问答交互。

(6) Web Screenshot2Code: HTML code and corresponding web screenshots are readily available, providing rich information regarding the structure and interactivity of web elements (e.g., whether an icon is clickable or hoverable). We investigate whether leveraging such accessible data can enhance the performance of GUI-Agents on downstream tasks. Specifically, we construct the training dataset by randomly sampling 150K web screenshot-HTML code pairs from the Web2Code dataset (Yun et al., 2024).

(6) 网页截图转代码:HTML 代码及对应网页截图易于获取，提供了丰富的网页元素结构和交互性信息 (如图标是否可点击或悬停)。我们探讨利用此类数据是否能提升 GUI 代理在下游任务中的表现。具体而言，我们通过从 Web2Code 数据集 (Yun 等，2024) 随机抽取 15 万网页截图-HTML 代码对构建训练数据集。

(7) Non-GUI Agent Trajectories: With the rapid development of Embodied AI, a growing amount of non-GUI agent data is becoming available in domains beyond the web, such as household tasks (Shridhar et al., 2021). We investigate whether data from these domains can benefit GUI Agent tasks. Specifically, due to the limited availability of such data, we utilize all 51K samples from the AlfWorld subset of MAmmoTH-VL (Guo et al., 2024) as mid-training data for our experiments.

(7) 非图形界面代理轨迹: 随着具身人工智能 (Embodied AI) 的快速发展，越来越多非图形界面代理数据在网页之外的领域变得可用，例如家务任务 (Shridhar 等，2021)。我们探讨这些领域的数据是否能惠及图形界面代理任务。具体而言，由于此类数据的有限性，我们利用 MAmmoTH-VL(Guo 等，2024) 中 AlfWorld 子集的全部 51K 样本作为中期训练数据进行实验。

Text data is often more readily available and abundant compared to vision-language data, widely accessible through sources such as the internet. We investigate whether pure text data can enhance the capabilities of GUI Agents. For the text modality, our primary focus includes the following:

文本数据通常比视觉-语言数据更易获得且更丰富，广泛可通过互联网等渠道获取。我们研究纯文本数据是否能提升图形界面代理的能力。对于文本模态，我们的主要关注点包括以下内容:

(1) MathInstruct: Text-based math datasets (Cobbe et al., 2021) are commonly used to enhance the reasoning capabilities of models. We randomly sample 150K examples from the CoT category of MathInstruct (Yue et al., 2023) as mid-training data.

(1) MathInstruct: 基于文本的数学数据集 (Cobbe 等, 2021) 常用于增强模型的推理能力。我们从 MathInstruct(Yue 等，2023) 的 CoT 类别中随机抽取 15 万个样本作为中期训练数据。

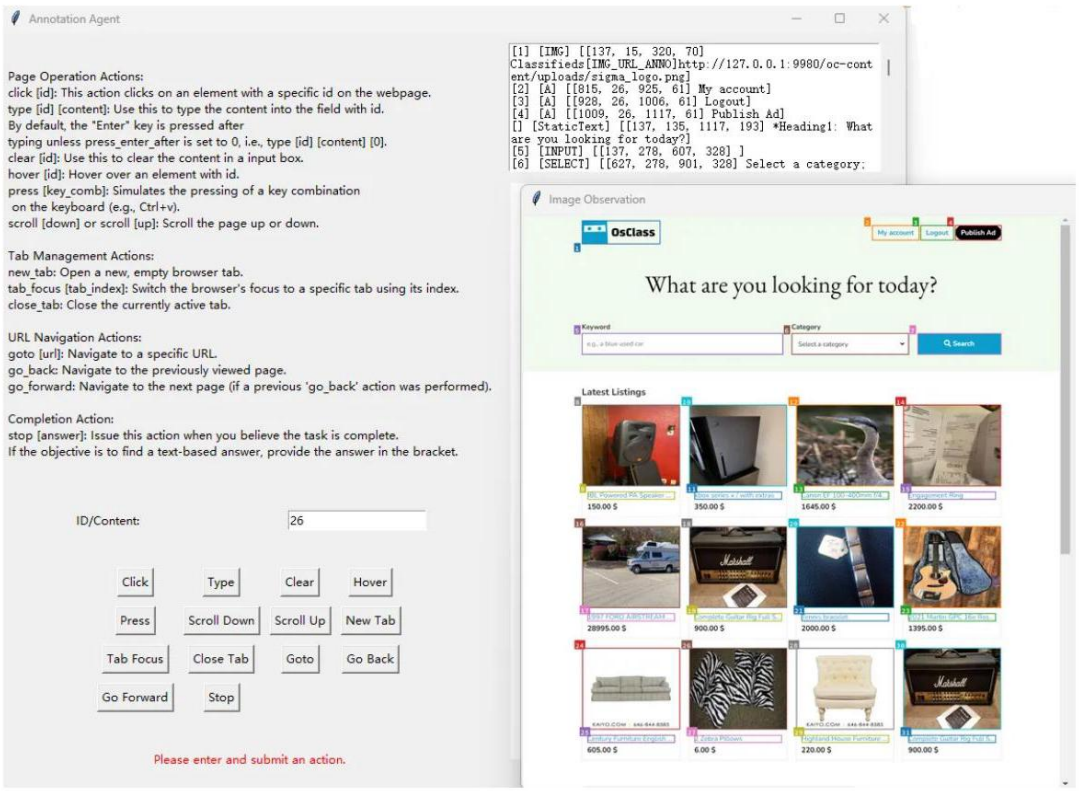

Figure 5: The annotation UI for VisualWebArena.

图 5:VisualWebArena 的标注用户界面。

(2) CodeI/O: CodeI/O (Li et al., 2025) is a novel approach that transforms code-based reasoning patterns into natural language formats to enhance the reasoning capabilities of Large Language Models. We randomly sample 150K examples from the CodeI/O dataset.

(2) CodeI/O:CodeI/O(Li 等，2025) 是一种新颖方法，将基于代码的推理模式转化为自然语言格式，以增强大型语言模型 (Large Language Models) 的推理能力。我们从 CodeI/O 数据集中随机抽取 15 万个样本。

(3) Web Knowledge Base: Text-based trajectories can be synthesized using methods like (Ou et al., 2024; Xu et al., 2024a), which leverage tutorial knowledge. However, generating multi-modal data at scale remains challenging. In this work, we explore whether text-based trajectory data can benefit vision-based GUI agents. Specifically, we utilize 100K trajectories from the Synatra dataset (Ou et al., 2024) and randomly sample 50K trajectories from the AgentTrek dataset (Xu et al., 2024a), both of which are web-domain datasets that may potentially enhance GUI-Agent performance on web-based tasks.

(3) 网络知识库: 可通过 (Ou 等，2024；Xu 等，2024a) 等方法合成基于文本的轨迹，这些方法利用教程知识。然而，大规模生成多模态数据仍具挑战性。在本工作中，我们探索基于文本的轨迹数据是否能惠及基于视觉的图形界面代理。具体而言，我们利用 Synatra 数据集 (Ou 等，2024) 中的 10 万条轨迹，并从 AgentTrek 数据集 (Xu 等，2024a) 随机抽取 5 万条轨迹，这两个数据集均属于网页领域，可能提升图形界面代理在网页任务上的表现。

(4) Olympiad Math: NuminaMath (LI et al., 2024) is a comprehensive mathematical dataset comprising a wide range of problems, including exercises from Chinese high school mathematics curricula as well as competition problems from US and international mathematics olympiads. We specifically select the most challenging subset problems categorized under Olympiads within NuminaMath, and randomly sample 150K examples from this subset to highlight the complexity inherent in advanced mathematical reasoning.

(4) 数学奥林匹克:NuminaMath(LI 等，2024) 是一个综合数学数据集，涵盖广泛问题，包括中国高中数学课程练习题及美国和国际数学奥林匹克竞赛题。我们特别选取 NuminaMath 中归类为奥林匹克的最具挑战性的子集问题，并从中随机抽取 15 万个样本，以突出高级数学推理的复杂性。

C Details of GUI Trajectory Data

C 图形界面轨迹数据详情

High-quality GUI trajectory data is critical for enabling GUI agents to advance digital automation. However, due to constraints in application scenarios and annotation costs, existing high-quality trajectory data remains limited. To better adapt mid-trained models to GUI-specific tasks, we collected the following trajectory data:

高质量的图形界面轨迹数据对于推动图形界面代理实现数字自动化至关重要。然而，由于应用场景和标注成本的限制，现有高质量轨迹数据仍然有限。为更好地使中期训练模型适应图形界面特定任务，我们收集了以下轨迹数据:

(1) OS-Genesis: We leverage trajectory data from both mobile and web platforms provided by OS-Genesis (Sun et al., 2024b), which are automatically synthesized in reverse via an interaction-driven approach. To further improve the inference quality of these trajectories, we enhance them with step-wise reasoning using gpt- 40-mini (Hurst et al., 2024). Specifically, gpt-40-mini generates five sets of Chain of Thought data for each step and optimized them to ensure consistency with the corresponding action. If inconsistencies persist, the corresponding data will be discarded. Ultimately, approximately 4K high-quality single-step trajectory samples are collected for the web platform and 5K for the mobile platform.

(1) OS-Genesis: 我们利用 OS-Genesis(Sun 等，2024b) 提供的移动端和网页端轨迹数据，这些数据通过交互驱动方法自动逆向合成。为进一步提升轨迹推理质量，我们采用 gpt-40-mini(Hurst 等，2024) 进行逐步推理增强。具体而言，gpt-40-mini 为每一步生成五组思维链 (Chain of Thought) 数据，并优化以确保与对应动作一致。如仍存在不一致，则丢弃该数据。最终，网页端收集约 4 千条高质量单步轨迹样本，移动端收集约 5 千条。

(2) MM-Mind2Web: MM-Mind2Web (Zheng et al., 2024a) is the multi-modal extension of the Mind2Web dataset (Deng et al., 2023), designed for developing and evaluating generalist web agents capable of following natural language instructions to complete complex tasks on arbitrary websites. This dataset aligns each HTML document with its corresponding webpage screenshot from the Mind2Web raw dump, enabling joint modeling of structural and visual web information. We also employed GPT-40-mini to process the data, resulting in the CoT-enhanced MM-Mind2Web dataset with 21K annotated steps.

(2) MM-Mind2Web:MM-Mind2Web(Zheng 等，2024a) 是 Mind2Web 数据集 (Deng 等，2023) 的多模态扩展，旨在开发和评估能够遵循自然语言指令完成任意网站复杂任务的通用网页代理。该数据集将每个 HTML 文档与 Mind2Web 原始数据中的对应网页截图对齐，实现网页结构与视觉信息的联合建模。我们同样使用 GPT-40-mini 处理数据，生成了包含 2.1 万标注步骤的思维链增强 MM-Mind2Web 数据集。

(3) VisualWebArena: We randomly sample questions from different task template of the VisualWebArena (Koh et al., 2024a), and annotate 3,264 steps of data using our implemented annotation tools (Figure 5). To ensure the annotation is correct, each annotated trajectory will be replayed and is required to pass the task.

(3) VisualWebArena: 我们从 VisualWebArena(Koh 等，2024a) 不同任务模板中随机抽取问题，使用我们实现的标注工具 (图 5) 标注了 3,264 步数据。为确保标注正确，每条标注轨迹均会回放并需通过对应任务。

(4) Aguvis: Aguvis is a synthetic dataset of high-quality GUI agent trajectories that builds upon and enhances existing open-source datasets such as AitW (Rawles et al., 2023), GUI Odyssey (Lu et al., 2024a) and GUIAct (Chen et al., 2024). While these prior datasets typically include high-level goals, observations, and grounded actions, Aguvis enriches them by incorporating detailed intermediate reasoning and low-level action instructions. Leveraging a VLM, Aguvis generates step-by-step inner monologues comprising observation descriptions, agent thoughts, and fine-grained action instructions, enabling more advanced multi-modal reasoning and planning for GUI agents. We randomly sample 22K single-step reasoning data from Aguvis to support post-training.

(4) Aguvis:Aguvis 是一个高质量 GUI 代理轨迹的合成数据集，基于并增强了现有的开源数据集，如 AitW(Rawles 等，2023)、GUI Odyssey(Lu 等，2024a) 和 GUIAct(Chen 等，2024)。虽然这些先前的数据集通常包含高级目标、观察和具体动作，Aguvis 通过加入详细的中间推理和低级动作指令来丰富它们。利用视觉语言模型 (VLM)，Aguvis 生成逐步的内心独白，包括观察描述、代理思考和细粒度动作指令，从而支持更高级的多模态推理和 GUI 代理规划。我们从 Aguvis 中随机抽取了 2.2 万条单步推理数据以支持后期训练。

D Training Details

D 训练细节

Following state-of-the-art works (Xu et al., 2024b; Sun et al., 2024b; Gou et al., 2025), we use Qwen2-VL-7B-Instruct (Wang et al., 2024a) as our backbone model. During the mid-training experiments, the middle training stage and post-training stage are integrated under a single optimizer and learning rate schedule, which we find important empirically to stabilize the training process. For the scaling law experiments presented in Figure 3, we save checkpoints of the models trained on GUIMid as shown in Table 3 based on the amount of mid-training data trained. To evaluate the performance at intermediate points in the scaling law experiments (e.g., after training on 50K samples of mid-training data), we use the corresponding checkpoints and continue training with a cosine learning rate schedule that gradually reduces the learning rate to zero, starting from the learning rate value recorded at the checkpoint. Due to hardware constraints, we limit the batch size per device to 1 with a gradient accumulation of 4 . For the results of scaling to 300 K samples, we directly report the performance in Table 3.

遵循最先进的工作 (Xu 等,2024b;Sun 等,2024b;Gou 等,2025),我们采用 Qwen2-VL-7B-Instruct(Wang 等，2024a) 作为骨干模型。在中期训练实验中，中期训练阶段和后期训练阶段整合在单一优化器和学习率调度下，经验上发现这对稳定训练过程非常重要。对于图 3 中展示的规模定律实验，我们根据中期训练数据的训练量，保存了在 GUIMid 上训练的模型检查点，如表 3 所示。为了评估规模定律实验中间点的性能 (例如，在训练了 5 万条中期训练样本后)，我们使用相应的检查点，并继续采用余弦学习率调度，从检查点记录的学习率值开始，逐渐将学习率降至零。由于硬件限制，我们将每个设备的批量大小限制为 1，梯度累积为 4。对于扩展到 300 K 样本的结果，我们直接在表 3 中报告性能。

E Evaluation Details

E 评估细节

AndroidWorld. AndroidWorld evaluates autonomous agents in Android environments through 116 tasks across 20 real-world applications. Each task incorporates randomized parameters to generate diverse scenarios, enabling rigorous evaluation. Success rates (SR) are determined through system state inspection without modifying the original application source code. Due to application availability constraints, our final evaluation encompasses 111 tasks. We build GUI agents based on the M3A agent from AndroidWorld (Rawles et al.,

AndroidWorld。AndroidWorld 通过 20 个真实应用中的 116 个任务评估 Android 环境中的自主代理。每个任务包含随机参数以生成多样化场景,实现严格评估。成功率 (SR) 通过系统状态检查确定,无需修改原始应用源码。由于应用可用性限制, 我们最终评估涵盖了 111 个任务。我们基于 AndroidWorld 的 M3A 代理构建 GUI 代理 (Rawles 等,

Parameter	Value
Context Length	8192
Number of GPUs	8
Learning Rate	2×10^{-5}
Training Epochs	1.0
Batch Size Per Device	2
Gradient Accumulation	2
Learning Rate Scheduler	cosine
Warmup Ratio	0.05
Precision	bf16

参数	数值
上下文长度	8192
GPU 数量	8
学习率	2×10^{-5}
训练轮数	1.0
每设备批量大小	2
梯度累积	2
学习率调度器	余弦
预热比例	0.05
精度	bf16

Table 8: The training hyperparameters used in Table 3.

表 8: 表 3 中使用的训练超参数。

2024), drawing inspiration from the implementation of SeeAct-V agents (Gou et al., 2025), which rely solely on visual information by removing the SoM images and textual lists of elements from accessibility trees in the observations and converting element-based actions into pixel-level actions. Furthermore, to ensure the fairness and rigor of the experiment, we maintain consistency between the training and evaluation prompts by refraining from using deliberately crafted prompts targeting corner cases. This suggests that variations in agent performance can predominantly be attributed to the mid-training data.

2024 年)，借鉴 SeeAct-V 代理 (Gou 等，2025 年) 的实现，该代理仅依赖视觉信息，通过从观测中移除 SoM 图像和可访问性树中的元素文本列表，并将基于元素的操作转换为像素级操作。此外，为确保实验的公平性和严谨性，我们保持训练和评估提示的一致性，避免使用针对边缘案例的刻意设计提示。这表明代理性能的差异主要可归因于中期训练数据。

WebArena. We utilize the agentboard (Ma et al., 2024) version of WebArena (Zhou et al., 2023), which contains 245 cases sampled from the original benchmark, enhanced with progress labels that provide both success rates and progress rates-offering more granular insights into model capability development. To ensure evaluation accuracy and consistency, we deploy the Docker-based websites on our local infrastructure rather than using the public AWS setup, which can lead to cross-evaluation inaccuracies. Map-related tasks were excluded due to website configuration issues, leaving 211 tasks. For maximum reliability, we restart the environment after each evaluation to eliminate potential cross-contamination between test sessions.

WebArena。我们使用 agentboard(Ma 等，2024 年) 版本的 WebArena(Zhou 等，2023 年)，该版本包含从原始基准中采样的 245 个案例，增强了进度标签，提供成功率和进度率，能够更细致地洞察模型能力的发展。为确保评估的准确性和一致性，我们在本地基础设施上部署基于 Docker 的网站，而非使用可能导致交叉评估误差的公共 AWS 环境。由于网站配置问题，排除了地图相关任务，剩余 211 个任务。为最大限度保证可靠性，我们在每次评估后重启环境，以消除测试会话间的潜在交叉污染。

gpt-4o Baselines In the absence of 7B-parameter baselines with comparable data volume, we evaluate the strong model gpt-4o-2024-11-20 as a planning model in pure vision-based GUI scenarios, while employing UGround-V1-7B (Gou et al., 2025) as the grounding model. To ensure fair comparison, all models generate outputs using an

identical prompt as shown in Figure 10,11. Notably, SeeAct-V (Gou et al., 2025) demonstrates exceptional performance under these same experimental conditions. This superior performance can be attributed to its meticulously crafted prompt specifically designed for AndroidWorld, which addresses common edge cases and incorporates rules aligned with human intuition.

gpt-4o 基线在缺乏具有相当数据量的 7B 参数基线的情况下，我们在纯视觉 GUI 场景中将强模型 gpt-4o-2024-11-20 作为规划模型，同时采用 UGround-V1-7B(Gou 等，2025 年) 作为定位模型。为确保公平比较，所有模型均使用图 10、11 所示的相同提示生成输出。值得注意的是，SeeAct-V(Gou 等，2025 年) 在相同实验条件下表现卓越。这一优异表现归因于其为 AndroidWorld 精心设计的提示，专门处理常见边缘案例并融入符合人类直觉的规则。

All the experiments are conducted with temperate as 0.0, top_p as 1.0, max context length 8192.

所有实验均在温度为 0.0、top_p 为 1.0、最大上下文长度 8192 的条件下进行。

F Prompt

F 提示

We list our prompt below:

我们列出如下提示:

1. The prompt for generating Chain-of-Thought on OS-Genesis (Mobile) trajectories.

1. 用于生成 OS-Genesis(移动端) 轨迹的思维链提示。

2. The prompt for generating Chain-of-Thought on OS-Genesis (Web) trajectories.

2. 用于生成 OS-Genesis(网页端) 轨迹的思维链提示。

3. The prompt for generating Chain-of-Thought on VisualWebArena trajectories.

3. 用于生成 VisualWebArena 轨迹的思维链提示。

4. The prompt for generating Chain-of-Thought on Mind2Web trajectories.

4. 用于生成 Mind2Web 轨迹的思维链提示。

5. The prompt for evaluation on the AndroidWorld.

5. 用于 AndroidWorld 评估的提示。

6. The prompt for evaluation on the WebArena.

6. 用于 WebArena 评估的提示。

Prompt for Generating CoT on OS-Genesis (Mobile) Trajectories

用于生成 OS-Genesis(移动端) 轨迹思维链的提示

You are a mobile agent. Please think step by step and perform a series of actions

你是一名移动端代理。请逐步思考并执行一系列操作

on an Android device to complete the task. At each stage, you will receive the current

在安卓设备上完成任务。在每个阶段，你将收到当前

screenshot, a record of previous actions, and a hint for the next correct step. Based on

截图，之前操作的记录，以及下一步正确操作的提示。基于

this information, decide the next action to take without mentioning the provided correct

这些信息，决定下一步操作，但不要提及提供的正确

answer hint.

答案提示。

##Available actions:

可用操作:

UI Operations:

用户界面操作:

- ‘click [element]‘: Click on an element.
- ‘click [element]‘: 点击某个元素。
- ’type [element] [value]‘: Type content into a field by ID.
- ‘type [element] [value]‘: 在指定 ID 的字段中输入内容。
- ‘scroll [element] [value]‘: Scroll the page or a specific element. The direction can be

- ‘scroll [element] [value]‘: 滚动页面或特定元素。方向可以是

’up’, ’down’, ’left’, or ’right’. Leave the element blank for scrolling the entire page.

“上”、“下”、“左”或“右”。若元素为空，则滚动整个页面。

- ‘go_back‘: Navigate back.

- ‘go_back‘: 返回上一页。

- ‘go_home‘: Navigate to the home screen.

- ‘go_home‘: 返回主屏幕。

- ’long_press [element]’: Long press on an element.

- ‘long_press [element]‘: 长按某个元素。

- ’enter’: Press the Enter key.

- ‘enter‘: 按下回车键。

- ‘open_app [app_name]‘: Open an app by name.

- ‘open_app [app_name]‘: 打开指定名称的应用。

- ‘wait [value]‘: Wait for the screen to update for [value] seconds.

- ‘wait [value]‘: 等待 [value] 秒以更新屏幕。

###Task finishing:

任务完成:

- ‘stop [value]‘: Stop the task with a goal status or answer. If you think the task

- ‘stop [value]‘: 以目标状态或答案停止任务。如果你认为任务

is completed or infeasible, set the status to ’success’ or ’infeasible’. If the task

已完成或不可行，将状态设置为“success”(成功) 或“infeasible”(不可行)。如果任务

requires an answer, provide the answer here.

需要答案，请在此提供答案。

###Instruction for the thought process:

思考过程指导:

1. Describe the situation in detail, focusing on the goal and the related visual cues

1. 详细描述当前截图中的情况，重点关注目标和相关的视觉线索，

in current screenshot. Ensure your reasoning aligns with the goal, predicting the most

确保你的推理与目标一致，基于截图和之前的操作预测最

suitable action based on the screenshot and previous actions.

合适的动作。

2. Aim to reason through the task as if solving it, rather than simply reflecting on the

2. 目标是像解决问题一样推理任务，而不仅仅是反思

labeled outcome.

标注的结果。

3. Conclude the action with the format below.

3. 用以下格式总结动作。

Finally, end your thinking process with the action below.

最后，用下面的动作结束你的思考过程。

In summary, the next action is:

总结，下一步动作是:

...

{

”Element Description”: ”Describe the element you want to interact with, including its

“元素描述”: 描述你想要交互的元素，包括其

identity, type (e.g., button, input field, dropdown, tab), and any visible text. Keep the

身份、类型 (例如按钮、输入框、下拉菜单、标签页) 及任何可见文本。描述简洁，控制在 30 字以内。如果有相似元素，提供区分细节。

description concise and under 30 words. If there are similar elements, provide details to

distinguish them.”,

”Action”: ”Select an action from the following options: {click, type, scroll, go_back,

” 操作”: ” 从以下选项中选择一个操作:{点击, 输入, 滚动, 返回,

go_home, long_press, enter, open_app, stop}. Choose one; do not leave it blank.”,

回到主页, 长按, 回车, 打开应用, 停止}。请选择一个，不要留空。”,

”Value”: ”Provide additional input based on the chosen action:

” 数值”: ” 根据所选操作提供额外输入:

- For ’type’: specify the input text.
  - 对于‘输入’: 指定输入文本。
- For ’scroll’: specify the direction (”up”, ”down”, ”left”, ”right”).
  - 对于‘滚动’: 指定方向 (“上”，“下”，“左”，“右”)。
- For ’open_app’: provide the app name in the format: app_name=”the name of the app”.
  - 对于‘打开应用’: 以格式 app_name=” 应用名称” 提供应用名称。
- For ’stop’: provide ”completed”, ”infeasible”, or the required answer.
  - 对于‘停止’: 提供“完成”，“不可行”或所需答案。
- For ’wait’: provide the waiting time in seconds, in the format: seconds=”5s”.
  - 对于‘等待’: 以格式 seconds=”5s” 提供等待时间 (秒)。
- For other actions: leave this field empty.”
  - 对于其他操作: 此字段留空。”

...

###Input:

输入:

Previous Actions: {previous_actions}

先前操作:{previous_actions}

Task: {intent}

任务:{intent}

Correct Action Hint: {correct_answer}

正确操作提示:{correct_answer}

Figure 6: Prompt for generating Chain-of-Thought on OS-Genesis (Mobile) trajectories.

图 6: 用于生成 OS-Genesis(移动端) 轨迹链式思维 (Chain-of-Thought) 的提示。

Prompt for Generating CoT on OS-Genesis (Web) Trajectories

生成 OS-Genesis(网页端) 轨迹链式思维的提示

Imagine that you are imitating humans performing web navigation for a task, step by step. At each stage,

设想你正在模仿人类逐步执行网页导航任务。在每个阶段,

you can see the webpage as humans do through a screenshot, and you know the previous actions based on recorded

你可以通过截图像人类一样查看网页，并且根据记录的历史

history, the current screenshot, and meta information about the current website. You need to decide on the next

当前截图和关于当前网站的元信息，了解之前的操作。你需要决定下一步

action to take.

要采取的操作。

I will provide you with the hint answer. Please do not mention the hint answer in your thought process and just

我会给你提示答案。请不要在思考过程中提及提示答案，只需

reason through the task as if solving it yourself, but make sure your answer is the same with the hint answer.

像自己解决问题一样推理，但确保你的答案与提示答案一致。

##Available actions:

可用操作:

Web Operations:

网页操作:

- ‘click [element]‘: Click on an element.

- ‘click [element]‘: 点击一个元素。

- - ‘type [element] [value]‘: Type content into a field by ID.

- - ‘type [element] [value]‘: 在指定 ID 的字段中输入内容。

- ‘clear [element]‘: Clear the content of an element.

- ‘clear [element]‘: 清除元素内容。

- ’hover [element]’: Hover over an element by ID.

- ‘hover [element]‘: 鼠标悬停在指定 ID 的元素上。

- ’press [value]’: Press a key combination (e.g., Ctrl+v).

- ‘press [value]‘: 按下一个按键组合 (例如 Ctrl+v)。

- ‘scroll [down]‘ or scroll [up]* Scroll the page.

- ‘scroll [down]‘ 或 ‘scroll [up]‘: 滚动页面。

Tab Management:

标签管理:

- ‘new_tab‘: Open a new tab.

- ‘new_tab‘: 打开新标签页。

- ‘tab_focus [tab_index]‘: Switch to a specific tab.

- ‘tab_focus [tab_index]‘: 切换到指定标签页。

- ‘close_tab‘: Close the current tab.

- ‘close_tab‘: 关闭当前标签页。

URL Navigation:

URL 导航:

- ’goto [url]‘: Navigate to a URL.

- ‘goto [url]‘: 导航到指定网址。

- ‘go_back‘: Go to the previous page.

- ‘go_back‘: 返回上一页。

- ‘go_forward‘: Go to the next page.

- ‘go_forward‘: 前进到下一页。

###Task finishing:

任务完成:

- ‘stop [answer]‘: Issue this action when you believe the task is complete.

- ‘stop [answer]‘: 当你认为任务完成时，执行此操作。

###Instruction for the thought process:

思考过程指导:

1. Describe the situation in detail, focusing on the goal and the related visual cues in current screenshot. Ensure

1. 详细描述当前情况，重点关注目标和当前截图中的相关视觉线索。确保

your reasoning aligns with the goal, predicting the most suitable action based on the screenshot and previous

你的推理与目标一致，基于截图和之前的操作预测最合适的动作。

actions.

2. Aim to reason through the task as if solving it, rather than simply reflecting on the labeled outcome.

2. 目标是像解决问题一样推理任务，而非仅仅反思标注的结果。

3. Conclude the action with the format below.

3. 以以下格式结束动作。

4. Do not mention the hint answer in your thought process. Instead, reason to the answer independently, but ensure

4. 不要在思考过程中提及提示答案。相反，要独立推理出答案，但确保

your answer matches the hint answer.

你的答案与提示答案一致。

Finally, end your thinking process with the action below.

最后，用以下动作结束你的思考过程。

In summary, the next action is:

总结，下一步操作是:

...

{

”Element Description”: ”Describe the element you want to interact with, including its identity, type (e.g., button,

” 元素描述”:” 描述您想要交互的元素，包括其身份、类型 (例如，按钮 (button)、

input field, dropdown, tab), and any visible text. Keep the description concise and under 30 words. If there are

输入框、下拉菜单、标签页)，以及任何可见文本。保持描述简洁，少于 30 个字。如果有

similar elements, provide details to distinguish them.”,

相似元素，请提供区分细节。”

”Action”: ”Select an action from the following options: {stop, click, type, scroll, go_back, go_forward, goto,

” 操作”:” 从以下选项中选择一个操作:{停止 (stop)、点击 (click)、输入 (type)、滚动 (scroll)、后退 (go_back)、前进 (go_forward)、跳转 (goto)、

clear, hover, press, new_tab, page_focus, close_tab}. Choose one action; do not leave this field blank.”,

清除 (clear)、悬停 (hover)、按键 (press)、新标签页 (new_tab)、页面聚焦 (page_focus)、关闭标签页 (close_tab)}。请选择一个操作；此字段不能为空。”

”Value”: ”Provide additional input based on the chosen action:

” 数值”:” 根据所选操作提供额外输入:”

- For ’click’: specify the element to click.
  - 对于“点击 (click)”: 指定要点击的元素。
- For ’type’: specify the input text.
  - 对于“输入 (type)”: 指定输入文本。
- For ’scroll’: specify the direction (”up”, ”down”).
  - 对于“滚动 (scroll)”: 指定方向 (“上 (up)”、“下 (down)”)。
- For ’goto’: specify the URL to navigate to.
  - 对于“跳转 (goto)”: 指定要导航的 URL。
- For ’clear’: leave this field empty.
  - 对于“清除 (clear)”: 此字段留空。
- For ’hover’: specify the element to hover over.
  - 对于“悬停 (hover)”: 指定要悬停的元素。
- For ’press’: specify the key combination to press.
  - 对于“按键 (press)”: 指定要按下的按键组合。
- For ’stop’: provide one of the following: ”completed”, ”infeasible”, or the required answer.
  - 对于“停止 (stop)”: 提供以下之一:“完成 (completed)”、“不可行 (infeasible)”或所需答案。
- For ’wait’: provide the waiting time in seconds, in the format: seconds=”5s”.
  - 对于“等待 (wait)”: 以格式 seconds=”5s” 提供等待时间 (秒)。
- For all other actions: leave this field empty.”

- 对于所有其他操作: 请保持此字段为空。

}

###Input:

输入:

Previous Actions: {previous_actions}

先前操作:{previous_actions}

Task: {intent}

任务:{intent}

Correct Action Hint: {correct_answer}

正确操作提示:{correct_answer}

Figure 7: Prompt for generating Chain-of-Thought on OS-Genesis (Web) trajectories.

图 7: 用于生成 OS-Genesis(Web) 轨迹链式思考的提示。

Prompt for Generating CoT on VisualWebArena Trajectories

用于生成 VisualWebArena 轨迹链式思考的提示

Imagine that you are imitating humans performing web navigation for a task, step by step. At each stage,

假设你正在模仿人类逐步执行网页导航任务。在每个阶段,

you can see the webpage as humans do through a screenshot, and you know the previous actions based on recorded

你可以通过截图像人类一样看到网页，并且你知道基于记录的

history, the current screenshot, and meta information about the current website. You need to decide on the next

历史、当前截图以及当前网站的元信息的先前操作。你需要决定下一步

action to take.

要采取的操作。

I will provide you with the hint answer. Please do not mention the hint answer in your thought process and just

我会给你提示答案。请不要在思考过程中提及提示答案，只需

reason through the task as if solving it yourself, but make sure your answer is the same with the hint answer.

像自己解决问题一样推理，但确保你的答案与提示答案一致。

##Available actions:

可用操作:

Web Operations:

网页操作:

- ‘click [element]‘: Click on an element.
- ‘click [element]‘: 点击一个元素。
- ’type [element] [value]‘: Type content into a field by ID.
- ’type [element] [value]‘: 通过 ID 在字段中输入内容。
- ‘clear [element]‘: Clear the content of an element.
- ‘clear [element]‘: 清除元素内容。
- ’hover [element]’: Hover over an element by ID.
- ’hover [element]’: 将鼠标悬停在指定 ID 的元素上。
- ’press [value]‘: Press a key combination (e.g., Ctrl+v).
- ’press [value]‘: 按下组合键 (例如 Ctrl+v)。
- ‘scroll [down]‘ or ‘scroll [up]‘: Scroll the page.
- ‘scroll [down]‘ 或 ‘scroll [up]‘: 滚动页面。

Tab Management:

标签页管理:

- ‘new_tab‘: Open a new tab.

- ‘new_tab‘: 打开新标签页。

- ‘page_focus [tab_index]‘: Switch to a specific tab.

- ‘page_focus [tab_index]‘: 切换到指定标签页。

- ‘close_tab‘: Close the current tab.

- ‘close_tab‘: 关闭当前标签页。

URL Navigation:

URL 导航:

- ’goto [url]’: Navigate to a URL.

- ’goto [url]’: 跳转到指定 URL。

- ‘go_back‘: Go to the previous page.

- ‘go_back‘: 返回上一页。

- ‘go_forward‘: Go to the next page.

- ‘go_forward‘: 前进到下一页。

###Task finishing:

任务完成:

- ‘stop [answer]‘: Issue this action when you believe the task is complete.

- ‘stop [answer]‘: 当你认为任务完成时执行此操作。

###Instruction for the thought process:

思考过程指引:

1. Describe the situation in detail, focusing on the goal and the related visual cues in the current screenshot.

1. 详细描述当前截图中的情况，重点说明目标和相关的视觉提示。

Ensure your reasoning aligns with the goal, predicting the most suitable action based on the screenshot and previous

确保你的推理与目标一致，基于截图和之前的操作预测最合适的动作。

actions.

操作。

2. Aim to reason through the task as if solving it, rather than simply reflecting on the labeled outcome.

2. 目标是像解决问题一样推理任务，而不仅仅是反思标注的结果。

3. Conclude the action with the format below.

3. 以以下格式总结动作。

4. Do not mention the hint answer in your thought process, but make sure your answer is the same with the hint answer.

4. 不要在思考过程中提及提示答案，但确保你的答案与提示答案一致。

Finally, end your thinking process with the action below.

最后，用以下动作结束你的思考过程。

In summary, the next action is:

总之，下一步动作是:

...

{

”Element Description”: ”Describe the element you want to interact with, including its identity, type (e.g., button,

” 元素描述”: ” 描述你想要交互的元素，包括其身份、类型 (例如按钮、输入框、下拉菜单、标签页) 及任何可见文本。描述简洁，控制在 30 字以内。如有相似元素，提供区分细节。”,

input field, dropdown, tab), and any visible text. Keep the description concise and under 30 words. If there are

输入框、下拉菜单、标签页)，以及任何可见文本。保持描述简洁，少于 30 个字。如果有

similar elements, provide details to distinguish them.”,

相似元素，提供区分细节。”,

”Action”: ”Select an action from the following options: {stop, click, type, scroll, go_back, go_forward, goto,

” 动作”: ” 从以下选项中选择一个动作:{停止 (stop), 点击 (click), 输入 (type), 滚动 (scroll), 后退 (go_back), 前进 (go_forward), 跳转 (goto),

clear, hover, press, new_tab, page_focus, close_tab}. Choose one action; do not leave this field blank.”,

清除 (clear), 悬停 (hover), 按压 (press), 新标签页 (new_tab), 页面聚焦 (page_focus), 关闭标签页 (close_tab)}。选择一个动作，不能为空。”,

”Value”: ”Provide additional input based on the chosen action:

” 数值”: ” 根据所选动作提供额外输入:

- For ’click’: specify the element to click.
- 对于“点击 (click)”: 指定要点击的元素。
- For ’type’: specify the input text.
- 对于“输入 (type)”: 指定输入的文本。
- For ’scroll’: specify the direction (”up”, ”down”).
- 对于“scroll”: 指定方向 (“up”，“down”)。
- For ’goto’: specify the URL to navigate to.
- 对于“goto”: 指定要导航的 URL。
- For ’clear’: leave this field empty.
- 对于“clear”: 此字段留空。
- For ’hover’: specify the element to hover over.
- 对于“hover”: 指定要悬停的元素。
- For ’press’: specify the key combination to press.
- 对于“press”: 指定要按下的按键组合。
- For ’page_focus’: specify the tab index to switch to.
- 对于“page_focus”: 指定要切换的标签索引。

- For 'stop': provide one of the following: "completed", "infeasible", or the required answer.

- 对于“stop”: 提供以下之一:“completed”、“infeasible”或所需答案。

- For 'wait': provide the waiting time in seconds, in the format: seconds="5s".

- 对于“wait”: 以格式 seconds="5s" 提供等待时间 (秒)。

- For all other actions: leave this field empty."

- 对于所有其他操作: 此字段留空。

}

###Input:

输入:

Current URL: {url}

当前 URL:{url}

Previous Actions: {previous_actions}

之前的操作:{previous_actions}

Task: {intent}

任务:{intent}

Hint_Action: {hint_action}

提示操作:{hint_action}

Figure 8: Prompt for generating Chain-of-Thought on VisualWebArena trajectories.

图 8: 用于生成 VisualWebArena 轨迹链式思维 (Chain-of-Thought) 的提示。

Prompt for generating CoT on Mind2Web trajectories

用于生成 Mind2Web 轨迹链式思维 (CoT) 的提示

You are a smart and helpful visual assistant that is well-trained to manipulate

你是一个聪明且乐于助人的视觉助手，经过良好训练，能够操作

websites. Your task is to navigate and take action on the current screen step-by-step to

网站。你的任务是逐步浏览并在当前屏幕上执行操作，

complete the user request.

以完成用户请求。

I will provide you with the hint answer. Please do not mention the hint answer in your

我会提供给你提示答案。请不要在思考过程中提及提示答案，

thought process and just reason through the task as if solving it yourself, but make sure

而是像自己解决问题一样推理，但确保

your answer is the same with the hint answer.

你的答案与提示答案一致。

##Instructions:

操作说明:

- You will be provided with screenshots and website information.
- 你将获得截图和网站信息。
- Review your previous actions to determine the next steps. Go back to previous status if
- 回顾你之前的操作以确定下一步。如果必要，可以回到之前的状态。

necessary.

- Pay close attention to all specific requirements of the task.
- 密切关注任务的所有具体要求。

##Analysis Guidelines

分析指南

###Previous Actions Analysis:

之前操作分析:

- You should analyze the previous actions and the current status of the task.

- 你应分析之前的操作和任务的当前状态。

###Screenshot Description:

截图描述:

- You should describe all the screenshot in detail, especially the interactive elements,

- 你应详细描述所有截图，特别是交互元素，

such as buttons, search bars, and dropdown lists.

例如按钮、搜索栏和下拉列表。

###Sub-task Planning:

子任务规划:

- Analyze the task status based on the observation and past actions and detail a reasonable

- 根据观察和过去的操作分析任务状态，并详细制定合理的

future action plan to accomplish the user request.

未来行动计划以完成用户请求。

- You should carefully check **ALL THE SPECIFIC REQUIREMENTS** to make the plan.

- 你应仔细检查 ** 所有具体要求 ** 以制定计划。

- You MUST check whether the last action is conducted successfully by analyzing the current

- 你必须通过分析当前截图检查上一次操作是否成功执行。

screenshot.

截图。

###Critical Analysis and Reflection:

关键分析与反思:

- Check whether the history actions have accomplished the user request.

• 检查历史操作是否已完成用户请求。

• Critique the past actions and make a decision on the next action, and decide whether to

• 批判过去的操作并决定下一步行动，判断是否需要

backtrack to the previous steps with actions like: go back, goto url; scroll up:

回溯到前一步操作，如: 返回、跳转网址、向上滚动:

• Assess the feasibility of the current sub-task and the overall task, and decide whether

• 评估当前子任务及整体任务的可行性，并决定是否

to modify the plan.

修改计划。

Finally, end your thinking process with the action below.

最后，用以下操作结束你的思考过程。

In summary, the next action is:

总结，下一步操作是:

...

{

”Element Description”: ”Describe the element you want to interact with, including its

” 元素描述”:” 描述你想要交互的元素，包括其

identity, type (e.g., button, input field, dropdown, tab), and any visible text. Keep the

身份、类型 (例如按钮、输入框、下拉菜单、标签) 及任何可见文本。保持

description concise and under 30 words. If there are similar elements, provide details to

描述简洁，控制在 30 字以内。如有相似元素，提供细节以

distinguish them.”,

区分它们。

”Action”: ”Select an action from the following options: {click, type, scroll, go_back,

” 操作”: ” 从以下选项中选择一个操作:{点击 (click)、输入 (type)、滚动 (scroll)、返回 (go_back)、

go_home, long_press, enter, open_app, stop}. Choose one; do not leave it blank.",

回首页 (go_home)、长按 (long_press)、回车 (enter)、打开应用 (open_app)、停止 (stop)}。请选择一项，不能为空。”

"Value": "Provide additional input based on the chosen action:

" 值": " 根据所选操作提供额外输入:

- For 'type': specify the input text.
  - 对于“输入 (type)”：指定输入文本。
- For 'scroll': specify the direction ("up", "down", "left", "right").
  - 对于“滚动 (scroll)”：指定方向 (“上 (up)”、“下 (down)”、“左 (left)”、“右 (right)”)。
- For 'open_app': provide the app name in the format: app_name="the name of the app".
  - 对于“打开应用 (open_app)”：以格式 app_name=" 应用名称" 提供应用名。
- For 'stop': provide "completed", "infeasible", or the required answer.
  - 对于“停止 (stop)”：提供“完成 (completed)”、“不可行 (infeasible)”或所需答案。
- For 'wait': provide the waiting time in seconds, in the format: seconds="5s".
  - 对于“等待 (wait)”：以格式 seconds="5s" 提供等待时间 (秒)。
- For other actions: leave this field empty."
  - 对于其他操作: 此字段留空。"

}

###Input:

输入:

Task: {task}

任务: {task}

Previous Actions: {previous actions}

先前操作: {previous actions}

Hint Answer: {hint_answer}

提示答案: {hint_answer}

Figure 9: Prompt for generating Chain-of-Thought on Mind2Web trajectories.

图 9: 用于生成 Mind2Web 轨迹链式思维的提示。

Evaluation Prompt for Mobile Tasks

移动任务评估提示

<image>
You are a mobile agent. You need to perform a series of actions to complete a task on

你是一个移动代理。你需要在

Android, step by step. At each step, you are provided with the current screenshot and

Android 系统上逐步执行一系列操作以完成任务。每一步，你会获得当前截图和

previous actions you have taken. You need to decide on the next action to take.

之前执行的操作。你需要决定下一步的操作。

##Available actions:

可用操作:

UI Operations:

用户界面操作:

- ‘click [element]‘: Click on an element.

- ‘click [element]‘: 点击一个元素。

- ’type [element] [value]‘: Type content into a field by ID.

- ‘type [element] [value]‘: 在指定 ID 的字段中输入内容。

- ‘scroll [element] [value]‘: Scroll the page or a specific element. The direction can be

- ‘scroll [element] [value]‘: 滚动页面或特定元素。方向可以是

’up’, ’down’, ’left’, or ’right’. Leave the element blank for scrolling the entire page.

“上”、“下”、“左”或“右”。若元素为空，则滚动整个页面。

- ’go_back’: Navigate back.

- ‘go_back‘: 返回上一页。

- ‘go_home‘: Navigate to the home screen.

- ‘go_home‘: 返回主屏幕。

- ‘long_press [element]‘: Long press on an element.

- ‘long_press [element]‘: 长按一个元素。

- ’enter’: Press the Enter key.

- ‘enter‘: 按下回车键。

- ‘open_app [app_name]‘: Open an app by name.

- ‘open_app [app_name]‘: 通过名称打开应用。

- ‘wait [value]‘: Wait for the screen to update for [value] seconds.

- ‘wait [value]‘: 等待屏幕更新 [value] 秒。

###Task finishing:

任务完成:

- ‘stop [value]‘: Stop the task with a goal status or answer. If you think the task is

- ‘stop [value]‘: 以目标状态或答案停止任务。如果你认为任务是

completed or infeasible, set the status to ’successful’ or ’infeasible’. If the task

完成或不可行时，将状态设置为“成功”或“不可行”。如果任务

requires an answer, provide the answer here.

需要一个答案，请在此提供答案。

Please provide your detailed thought process and specify the action you intend to

请详细说明您的思考过程并指明您打算采取的行动

perform. The action should include a description of the element to be operated on, the

执行。该操作应包括对要操作元素的描述，

type of action, and the corresponding value, formatted as follows:

操作类型及其对应的值，格式如下:

...

{

”Element Description”: ”Describe the element you want to interact with.”,

” 元素描述”: ” 描述您想要交互的元素。”,

”Action”: ”Select an action from the available options, do not leave it blank.”,

” 操作”: ” 从可用选项中选择一个操作，不要留空。”,

”Value”: ”Provide a value only if the action requires it.”

“值”: 仅在操作需要时提供值。

...

###Input:

输入:

Previous Actions: {previous_actions}

先前操作:{previous_actions}

Task: {intent}

任务:{intent}

Figure 10: Evaluation prompt for mobile tasks.

图 10: 移动任务的评估提示。

Evaluation Prompt for Web Tasks

网络任务评估提示

<image>
Imagine that you are imitating humans performing web navigation for a task, step by step.

想象你正在模仿人类逐步执行网页导航任务。

At each stage, you can see the webpage as humans do through a screenshot, and you know the

在每个阶段，你可以通过截图像人类一样看到网页，并且你知道

previous actions based on recorded history, the current screenshot, and meta information

基于记录的历史、当前截图和当前网站的元信息的先前操作。

about the current website. You need to decide on the next action to take.

你需要决定下一步的操作。

##Available actions:

可用操作:

Web Operations:

网页操作:

- ‘click [element]‘: Click on an element.
- ‘click [element]‘: 点击一个元素。
- ’type [element] [value]‘: Type content into a field by ID.
- ‘type [element] [value]‘: 在指定 ID 的字段中输入内容。
- ‘clear [element]‘: Clear the content of an element.
- ‘clear [element]‘: 清除元素内容。
- ‘hover [element]‘: Hover over an element by ID.
- ‘hover [element]‘: 鼠标悬停在指定 ID 的元素上。
- ’press [value]‘: Press a key combination (e.g., Ctrl+v).

- ‘press [value]‘: 按下一个按键组合 (例如 Ctrl+v)。

- ‘scroll [down]‘ or ‘scroll [up]‘: Scroll the page.

- ‘scroll [down]‘ 或 ‘scroll [up]‘: 滚动页面。

Tab Management:

标签页管理:

- ‘new_tab‘: Open a new tab.

- ‘new_tab‘: 打开新标签页。

- ’page_focus [tab_index]‘: Switch to a specific tab.

- ‘page_focus [tab_index]‘: 切换到指定标签页。

- ‘close_tab‘: Close the current tab.

- ‘close_tab‘: 关闭当前标签页。

URL Navigation:

URL 导航:

- ‘goto [url]‘: Navigate to a URL.

- ‘goto [url]‘: 导航到指定的 URL。

- ‘go_back‘: Go to the previous page.

- ‘go_back‘: 返回上一页。

- ‘go_forward‘: Go to the next page.

- ‘go_forward‘: 前进到下一页。

###Task finishing:

任务完成:

- ’stop [answer]‘: Issue this action when you believe the task is complete (the value

• ‘stop [answer]‘: 当你认为任务完成时执行此操作 (值可以是成功、不可行或所需答案)。

could be successful, infeasible, or the required answer).

Please provide your detailed thought process and specify the action you intend to

请提供详细的思考过程并说明你打算执行的操作。操作应包括要操作元素的描述、

perform. The action should include a description of the element to be operated on, the

操作类型及相应的值，格式如下:

type of action, and the corresponding value, formatted as follows:

...

{

”Element Description”: ”Describe the element you want to interact with.”,

” 元素描述”:” 描述你想要交互的元素。”,

”Action”: ”Select an action from the available options. Choose one; do not leave it

” 操作”:” 从可用选项中选择一个操作。请选择一个，不要留空。”,

blank.”,

”Value”: ”Provide a value only if the action requires it.”

” 值”:” 仅当操作需要时提供值。”

}

###Input:

输入:

Current URL: {url}

当前 URL:{url}

Previous Actions: {previous_actions}

先前操作:{previous_actions}

Task: {intent}

任务:{intent}

Figure 11: Evaluation prompt for web tasks.

图 11: 网页任务的评估提示。