GUI Agents: A Survey

GUI 代理: 综述

Dang Nguyen [1*] Jian Chen [2] , Yu Wang [3] , Gang Wu [4] , Namyong Park, Zhengmian Hu[4] , Hanjia Lyu [5] , Junda Wu [6] , Ryan Aponte [7] , Yu Xia [6] , Xintong Li [6] , Jing Shi [4] , Hongjie Chen [8] , Viet Dac Lai [4] , Zhouhang Xie [6] , Sungchul Kim [4] , Ruiyi Zhang [4] , Tong Yu [4] , Mehrab Tanjim [4] , Nesreen K. Ahmed [9] , Puneet Mathur [4] , Seunghyun Yoon [4] , Lina Yao [10] , Jihyung Kil [4] , Branislav Kveton [4] , Thien Huu Nguyen [3] , Trung Bui [4] , Tianyi Zhou [1] , Ryan A. Rossi [4] , Franck Dernoncourt [4]

Dang Nguyen [1*] Jian Chen [2] , Yu Wang [3] , Gang Wu [4] , Namyong Park, Zhengmian Hu[4] , Hanjia Lyu [5] , Junda Wu [6] , Ryan Aponte [7] , Yu Xia [6] , Xintong Li [6] , Jing Shi [4] , Hongjie Chen [8] , Viet Dac Lai [4] , Zhouhang Xie [6] , Sungchul Kim [4] , Ruiyi Zhang [4] , Tong Yu [4] , Mehrab Tanjim [4] , Nesreen K. Ahmed [9] , Puneet Mathur [4] , Seunghyun Yoon [4] , Lina Yao [10] , Jihyung Kil [4] , Branislav Kveton [4] , Thien Huu Nguyen [3] , Trung Bui [4] , Tianyi Zhou [1] , Ryan A. Rossi [4] , Franck Dernoncourt [4]

[1] University of Maryland, [2] State University of New York at Buffalo, [3] University of Oregon, [4] Adobe Research, [5] University of Rochester, [6] University of California, San Diego, [7] Carnegie Mellon University, [8] Dolby Labs, [9] Cisco Research, [10] University of New South Wales

[1] 马里兰大学, [2] 纽约州立大学布法罗分校, [3] 俄勒冈大学, [4] Adobe 研究院, [5] 罗切斯特大学, [6] 加州大学圣地亚哥分校, [7] 卡内基梅隆大学, [8] Dolby 实验室, [9] 思科研究院, [10] 新南威尔士大学

Abstract

摘要

Graphical User Interface (GUI) agents, powered by Large Foundation Models, have emerged as a transformative approach to automating human-computer interaction. These agents autonomously interact with digital systems or software applications via GUIs, emulating human actions such as clicking, typing, and navigating visual elements across diverse platforms. Motivated by the growing interest and fundamental importance of GUI agents, we provide a comprehensive survey that categorizes their benchmarks, evaluation metrics, architectures, and training methods. We propose a unified framework that delineates their perception, reasoning, planning, and acting capabilities. Furthermore, we identify important open challenges and discuss key future directions. Finally, this work serves as a basis for practitioners and researchers to gain an intuitive understanding of current progress, techniques, benchmarks, and critical open problems that remain to be addressed.

图形用户界面 (GUI) 代理，借助大型基础模型 (Large Foundation Models)，已成为自动化人机交互的变革性方法。这些代理通过 GUI 自主与数字系统或软件应用交互，模拟人类的点击、输入和导航等操作，跨越多种平台。鉴于 GUI 代理日益增长的关注度及其基础性重要性，我们提供了一份全面的综述，分类介绍了其基准测试、评估指标、架构和训练方法。我们提出了一个统一框架，阐明其感知、推理、规划和执行能力。此外，我们识别了重要的开放挑战并讨论了关键的未来方向。最后，本工作为从业者和研究人员提供了直观理解当前进展、技术、基准及亟待解决的关键问题的基础。

1 Introduction

1 引言

Large Foundation Models (LFMs) have significantly transformed both the landscape of AI research and day-to-day life (Bommasani et al., 2022; Kapoor et al., 2024; Schneider et al., 2024; Naveed et al., 2023; Wang et al., 2024d). Recently, we have witnessed a paradigm shift from using LFMs purely as conversational chatbots (Touvron et al., 2023; Chiang et al., 2023; Dam et al., 2024) to employing them for performing actions and automating useful tasks (Wang et al., 2024b; Zhao et al., 2023; Yao et al., 2023; Shinn et al., 2023; Shen et al., 2024b; Cheng et al., 2024c). In this direction, one approach stands out: leveraging LFMs to interact with digital systems, such as desktops and mobile phones, or software applications such as a web browser, through Graphical User Interfaces (GUIs) in the same way humans do, for example, by controlling the mouse and keyboard to interact with visual elements displayed on a device's monitor (Iong et al., 2024; Hong et al., 2023; Lu et al., 2024; Shen et al., 2024a).

大型基础模型 (LFMs) 显著改变了人工智能研究领域和日常生活的格局 (Bommasani 等,2022;Kapoor 等，2024；Schneider 等，2024；Naveed 等，2023；Wang 等，2024d)。近期，我们见证了从将 LFMs 纯粹用作对话聊天机器人 (Touvron 等，2023；Chiang 等，2023；Dam 等，2024) 向利用其执行操作和自动化有用任务的范式转变 (Wang 等，2024b；Zhao 等，2023；Yao 等，2023；Shinn 等，2023；Shen 等，2024b；Cheng 等，2024c)。在这一方向上，一种方法尤为突出: 利用 LFMs 通过图形用户界面 (GUI) 与数字系统 (如桌面和手机) 或软件应用 (如网页浏览器) 交互，模仿人类通过控制鼠标和键盘与设备显示的视觉元素互动的方式 (Iong 等，2024；Hong 等，2023；Lu 等，2024；Shen 等，2024a)。

This approach holds great potential, as GUIs are ubiquitous across almost all computer devices that humans interact with in their work and daily lives. However, deploying LFMs in such environments poses unique challenges, such as dynamic layouts, diverse graphical designs across different platforms, and grounding issues, for instance, fine-grained recognition of elements within a page that are often small, numerous, and scattered (Liu et al., 2024b). Despite these challenges, many early efforts have shown significant promise (Lin et al., 2024; Cheng et al., 2024a), and growing interest from major players in the field is becoming evident [1] .

这种方法具有巨大的潜力，因为图形用户界面 (GUI) 在人类工作和日常生活中与之交互的几乎所有计算机设备上无处不在。然而，在这样的环境中部署大语言模型 (LFM) 带来了独特的挑战，例如动态布局、不同平台上多样的图形设计以及定位问题，例如对页面内通常小而多且分散的元素进行细粒度识别 (Liu 等人，2024b)。尽管存在这些挑战，但许多早期的努力已显示出显著的前景 (Lin 等人，2024；Cheng 等人，2024a)，并且该领域主要参与者的兴趣日益明显 [1] 。

Given the immense potential and rapid progress in this field, we propose a unified and systematic framework to categorize the various types of contributions within this space.

鉴于该领域的巨大潜力和快速进展，我们提出一个统一且系统的框架，对该领域内的各类贡献进行分类。

Organization of this Survey. We begin our survey by clearly defining the term "GUI Agent," followed by a formal definition of GUI agent tasks in Section 2. We then summarize different datasets and environments in Section 3 to provide readers a clearer picture of the kinds of problem settings currently available. We summarize various GUI agent architectural designs in Section 4, followed by different ways of training them in Section 5.

Lastly, we discuss open problems and future prospects of GUI agent research in Section 6.

本综述的结构。我们的综述首先明确定义“图形用户界面代理 (GUI Agent)”这一术语，然后在第 2 节中对图形用户界面代理任务进行正式定义。接着，我们在第 3 节中总结不同的数据集和环境，以便让读者更清楚地了解当前可用的问题设置类型。我们在第 4 节中总结各种图形用户界面代理的架构设计，随后在第 5 节中介绍训练它们的不同方法。最后，我们在第 6 节中讨论图形用户界面代理研究的开放性问题和未来前景。

---

*Corresponding author: dangmn@umd.edu

* 通讯作者:dangmn@umd.edu

[1] Anthropic, Google DeepMind, OpenAI

[1] 安普洛斯 (Anthropic)、谷歌深度思维 (Google DeepMind)、OpenAI

---

2 Preliminaries

2 预备知识

This section formally defines the term ”GUI agent” and presents a formalization of GUI agent tasks.

本节正式定义“图形用户界面代理”这一术语，并对图形用户界面代理任务进行形式化表述。

Definition 1 (GUI AGENT). An intelligent autonomous agent that interacts with digital platforms, such as desktops, or mobile phones, through their Graphical User Interface. It identifies and observes interactable visual elements displayed on the device’s screen and engages with them by clicking, typing, or tapping, mimicking the interaction patterns of a human user.

定义 1(图形用户界面代理)。一种智能自主代理，它通过图形用户界面与数字平台 (如桌面电脑或手机) 进行交互。它识别并观察设备屏幕上显示的可交互视觉元素，并通过点击、输入或轻触等方式与它们进行交互，模仿人类用户的交互模式。

Problem Formulation. GUI agent tasks involve an agent interacting with an environment in a sequential manner. The environment can generally be modeled as a Partially Observable Markov Decision Process (POMDP) (Sondik, 1971; Hauskrecht, 2000), defined by a tuple $(\mathcal{U}, \mathcal{A}, \mathcal{S}, \mathcal{O}, T)$, where $\mathcal{U}$ is the task space, $\mathcal{A}$ is the action space, $\mathcal{S}$ is the state space (not fully observable to the agent), $\mathcal{O}$ is the observation space, and $T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is a state transition function that maps a state-action pair to a probability distribution over subsequent states. A GUI agent is a policy $\pi : \Delta^{\mathcal{S}} \to \mathcal{A}$, where $\Delta^{\mathcal{S}}$ denotes the probability simplex over the state states. Most commonly, this is implemented using the entire history of past actions and observations. Given a task $u \in \mathcal{U}$, the agent proceeds through a sequence of actions to complete the task. At each step t , based on the history of past actions and observations, the policy π selects the next action $a \in \mathcal{A}$. The environment then transitions to a new state

$s' \in \mathcal{S}$ according to T . Depending on the environment's design, the agent may receive a reward $r = R(s, a, s')$, where $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is a reward function.

问题形式化。图形用户界面代理任务涉及代理以顺序方式与环境进行交互。环境通常可以建模为部分可观测马尔可夫决策过程 (POMDP)(桑迪克 (Sondik)，1971；豪斯克雷赫特 (Hauskrecht)，2000)，由一个元组 $(\mathcal{U}, \mathcal{A}, \mathcal{S}, \mathcal{O}, T)$ 定义，其中 $\mathcal{U}$ 是任务空间，$\mathcal{A}$ 是动作空间，$\mathcal{S}$ 是状态空间 (代理无法完全观测到)，$\mathcal{O}$ 是观测空间，$T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ 是一个状态转移函数，它将一个状态 - 动作对映射到后续状态的概率分布。图形用户界面代理是一个策略 $\pi : \Delta^{\mathcal{S}} \to \mathcal{A}$ ，其中 $\Delta^{\mathcal{S}}$ 表示状态上的概率单纯形。最常见的是，这是使用过去动作和观测的完整历史来实现的。给定一个任务 $u \in \mathcal{U}$ ，代理通过一系列动作来完成该任务。在每一步 t ，基于过去动作和观测的历史，策略 π 选择下一个动作 $a \in \mathcal{A}$ 。然后，环境根据 T 转移到一个新的状态 $s' \in \mathcal{S}$ 。根据环境的设计，代理可能会收到一个奖励 $r = R(s, a, s')$ ，其中 $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ 是一个奖励函数。

3 Benchmarks

3 基准测试

GUI agents are developed and evaluated on various platforms, including desktops, mobile phones, and web browser environments. This section summarizes benchmarks for all of these platform types.

图形用户界面代理在各种平台上进行开发和评估，包括桌面电脑、手机和网页浏览器环境。本节总结了所有这些平台类型的基准测试。

When evaluating GUI agents, it is crucial to distinguish between an environment and a dataset. A dataset is a static collection of data point, where each consists of several input features (e.g., a question, a screenshot of the environment, or the current state of the environment) and some output features (e.g., correct answers or actions to be taken). A dataset remains unchanged throughout the evaluation process. In contrast, an environment is an interactive simulation that represents a real-world scenario of interest. A GUI environment includes the GUI interface of a mobile phone or a desktop. Unlike datasets, environments are dynamic, actions taken within the environment can alter its state, hence, allowing modeling the problem as Markov Decision Processes (MDPs) or Partially Observable MDPs (POMDPs), with defined action, state, and observation spaces, and a state transition function.

在评估 GUI 代理时，区分环境 (environment) 和数据集 (dataset) 至关重要。数据集是静态的数据点集合，每个数据点包含若干输入特征 (例如问题、环境截图或环境当前状态) 和一些输出特征 (例如正确答案或应采取的动作)。数据集在整个评估过程中保持不变。相比之下，环境是一个交互式模拟，代表感兴趣的现实场景。GUI 环境包括手机或桌面的 GUI 界面。与数据集不同，环境是动态的，环境中的动作会改变其状态，因此可以将问题建模为马尔可夫决策过程 (MDPs) 或部分可观测马尔可夫决策过程 (POMDPs)，定义了动作空间、状态空间、观测空间及状态转移函数。

Another critical dimension of the existing benchmarks for GUI agents is the distinction between the open-world and closed-world assumptions. Closed-world datasets or environments presume that all necessary knowledge for solving a task is contained within the benchmark itself. In contrast, open-world benchmarks relax this constraint, allowing relevant information required to complete a task to exist outside the benchmark.

现有 GUI 代理基准测试的另一个关键维度是开放世界假设 (open-world) 与封闭世界假设 (closed-world) 的区分。封闭世界的数据集或环境假设解决任务所需的所有知识均包含在基准测试内部。相反，开放世界基准放宽了这一限制，允许完成任务所需的相关信息存在于基准之外。

We present a summary of existing GUI agent benchmarks in Table 1.

我们在表 1 中总结了现有的 GUI 代理基准测试。

3.1 Static Datasets

3.1 静态数据集

3.1.1 Closed-World Datasets.

3.1.1 封闭世界数据集

RUSS dataset introduces real-world instructions mapped to a domain-specific language (DSL) that enables agents to execute web-based tasks with high precision (Xu et al., 2021). Similarly, Mind2Web expands the task set to 2000 diverse tasks (Deng et al., 2023), and MT-Mind2Web adapts into conversational settings with multi-turn interactions (Deng et al., 2024). In contrast, TURK-INGBENCH focuses on common micro tasks in crowdsourcing platforms, featuring a rich mix of textual instructions, multi-modal elements, and complex layouts (Xu et al., 2024). Focusing on visual and textual interplay, VisualWebBench includes OCR, element grounding, and action prediction tasks, which require fine-grained multimodal understanding (Liu et al., 2024b). Similarly, ScreenSpot focuses on GUI grounding for clicking and typing directly from screenshots (Cheng et al., 2024b). Complementing this, WONDER-BREAD extends evaluation to business process management tasks, emphasizing workflow documentation and improvement rather than automation alone (Wornow et al., 2024). EnvDistraction dataset explores agent susceptibility to distractions in GUI environments, offering insights into faithfulness and resilience under cluttered and misleading contexts (Ma et al., 2024). NaviQAte introduces functionality-guided web application navigation, where tasks are framed as QA problems, pushing agents to extract actionable elements from multimodal inputs (Shahbandeh et al., 2024).

RUSS 数据集引入了映射到领域特定语言 (DSL) 的真实世界指令，使代理能够高精度执行基于网页的任务 (Xu 等, 2021)。类似地, Mind2Web 将任务集扩展到 2000 个多样化任务 (Deng 等, 2023), MT-Mind2Web 则适应多轮对话场景 (Deng 等，2024)。相比之下，TURK-INGBENCH 聚焦众包平台上的常见微任务, 包含丰富的文本指令、多模态元素和复杂布局 (Xu 等, 2024)。VisualWebBench 专注于视觉与文本的交互，涵盖 OCR、元素定位和动作预测任务，要求细粒度的多模态理解 (Liu 等，2024b)。同样，ScreenSpot 专注于从截图直接进行点击和输入的 GUI 定位 (Cheng 等，2024b)。WONDER-BREAD 则扩展评估至业务流程管理任务，强调工作流文档和改进，而非单纯自动化 (Wornow 等，2024)。EnvDistraction 数据集探讨代理在 GUI 环境中对干扰的敏感性，提供了在杂乱和误导性环境下的忠实度和鲁棒性见解 (Ma 等，2024)。NaviQAte 引入功能引导的网页应用导航，将任务框定为问答问题，推动代理从多模态输入中提取可操作元素 (Shahbandeh 等，2024)。

Evaluating on static closed-world datasets is particularly convenient, thanks to their lightweight and ease in

setting up compared to environments. They are also especially valuable for fine-grained evaluation, reproducibility, and comparing models under identical conditions. However, they lack the dynamism of real-world applications, as models are tested on fixed data rather than adapting to new inputs or changing scenarios.

在静态封闭世界数据集上评估尤其方便，因为它们相比环境更轻量且易于搭建。它们对于细粒度评估、结果复现以及在相同条件下比较模型尤为有价值。然而，它们缺乏现实应用的动态性，模型是在固定数据上测试，而非适应新输入或变化场景。

3.1.2 Open-World Datasets.

3.1.2 开放世界数据集

While most existing datasets are designed under the closed-world assumption, several datasets do not follow this paradigm. GAIA dataset tests agent integration diverse modalities and tools to answer real-world questions, often requiring web browsing or interaction with external APIs (Mialon et al., 2023). WebLINX emphasizes multi-turn dialogue for interactive web navigation on real-world sites, enhancing agents' adaptability and conversational skills (Lù et al., 2024).

尽管大多数现有数据集设计基于封闭世界假设，但也有若干数据集不遵循此范式。GAIA 数据集测试代理整合多模态和工具以回答现实问题，常需网页浏览或与外部 API 交互 (Mialon 等，2023)。WebLINX 强调多轮对话以实现真实网站的交互式网页导航，提升代理的适应性和对话能力 (Lù 等，2024)。

Evaluation on static open-world datasets balances the ease of evaluation with realism since the agents interact with real-world websites. However, due to the nature of real-world websites, they are often unpredictable and prone to changes, which makes it more challenging to reproduce and compare with prior methods.

在静态开放世界数据集上的评估兼顾了评估便利性和现实性，因为代理与真实网站交互。然而，由于真实网站的性质，它们常不可预测且易变，这使得复现和与先前方法比较更具挑战。

3.2 Interactive Environments

3.2 交互式环境

3.2.1 Closed-World Environments.

3.2.1 封闭世界环境

Closed-world interactive environments provide controlled and reproducible settings for evaluating agent capabilities. MiniWoB offers synthetic web tasks requiring interactions with webpages using mouse and keyboard inputs (Shi et al., 2017). It focuses on fundamental skills like button clicking and form filling, providing a baseline for evaluating low-level interaction. CompWoB extends MiniWoB with compositional tasks, requiring agents to handle multi-step workflows and generalize across task sequences (Furuta et al., 2023). This introduces dynamic

dependencies that reflect real-world complexity. WebShop simulates e-shopping tasks that challenge agents to navigate websites, process instructions, and make strategic decisions (Yao et al., 2022). WebArena advances realism with self-hosted environments across domains like e-commerce and collaborative tools, requiring agents to manage long-horizon tasks (Zhou et al., 2023b). VisualWebArena adds multimodal challenges, integrating visual and textual inputs for tasks like navigation and object recognition (Koh et al., 2024a). Shifting to enterprise settings, WorkArena evaluates agent performance in complex UI environments, focusing on knowledge work tasks in ServiceNow platform (Drouin et al., 2024). WorkArena++ extends this benchmark by introducing more challenging tasks (Boisvert et al., 2024). ST-WebAgentBench incorporates safety and trustworthiness metrics, assessing policy adherence and minimizing risky actions, critical for business deployment (Levy et al., 2024). Vide-oWebArena introduces long-context video-based tasks, requiring agents to understand instructional videos and integrate them with textual and visual data to complete tasks. It emphasizes memory retention and multimodal reasoning (Jang et al., 2024). In simulated desktop environments, OS-World (Xie et al., 2024) provides the first realistic operating system setting for evaluating multimodal GUI agents. Spider2-V (Cao et al., 2024) builds on this direction by targeting professional-level data science and engineering tasks. BrowserGym (Chezelles et al., 2024) develops a unified ecosystem consisting of seven web agent benchmarks for developing and evaluating web agents.

封闭世界交互环境提供了用于评估智能体能力的受控且可复现的设置。MiniWoB 提供了合成的网页任务，要求通过鼠标和键盘输入与网页交互 (Shi 等，2017)。它侧重于按钮点击和表单填写等基础技能，为评估低级交互提供了基线。CompWoB 在 MiniWoB 基础上扩展了组合任务，要求智能体处理多步骤工作流并在任务序列间进行泛化 (Furuta 等，2023)。这引入了反映现实复杂性的动态依赖关系。WebShop 模拟电子购物任务，挑战智能体在网站中导航、处理指令并做出策略决策 (Yao 等，2022)。WebArena 通过涵盖电商和协作工具等领域的自托管环境提升了现实感，要求智能体管理长时任务 (Zhou 等，2023b)。VisualWebArena 增加了多模态挑战，整合视觉和文本输入，用于导航和物体识别等任务 (Koh 等，2024a)。转向企业场景，WorkArena 评估智能体在复杂用户界面环境中的表现，聚焦 ServiceNow 平台上的知识工作任务 (Drouin 等，2024)。WorkArena++ 通过引入更具挑战性的任务扩展了该基准 (Boisvert 等，2024)。ST-WebAgentBench 融入安全性和可信度指标，评估策略遵守情况并最小化风险行为，这对业务部署至关重要 (Levy 等，2024)。VideoWebArena 引入基于长上下文的视频任务，要求智能体理解教学视频并将其与文本和视觉数据整合以完成任务，强调记忆保持和多模态推理 (Jang 等，2024)。在模拟桌面环境中，OS-World(Xie 等，2024) 提供了首个用于评估多模态图形用户界面智能体的真实操作系统环境。Spider2-V(Cao 等，2024) 沿此方向发展，针对专业级数据科学和工程任务。BrowserGym(Chezelles 等，2024) 开发了一个统一生态系统，包含七个网页智能体基准，用于开发和评估网页智能体。

Closed-world environments serve as evaluation platforms that mimic the dynamism of real-world environments while offering stability and reproducibility. However, setting up such benchmarks is often challenging, as they typically require considerable storage space and engineering skills.

封闭世界环境作为评估平台，模拟现实环境的动态性，同时提供稳定性和可复现性。然而，搭建此类基准通常具有挑战性，因为它们通常需要大量存储空间和工程技术。

3.2.2 Open-World Environments.

3.2.2 开放世界环境。

Open-world interactive environments challenge agents to navigate dynamic, real-world websites with evolving content and interfaces. WebVLN introduces a novel benchmark for vision-and-language navigation on websites, requiring agents to interpret visual and textual instructions to complete tasks such as answering user queries (Chen et al., 2024). It emphasizes multimodal reasoning by integrating HTML structure with rendered webpages, setting a foundation for realistic web navigation. WebVoyager leverages LLM to perform end-to-end navigation on 15 real websites with diverse tasks (He et al., 2024b). Its multimodal approach integrates screenshots and HTML content, enabling robust decision-making in dynamic online settings. AutoWebGLM optimizes web navigation through HTML simplification and reinforcement learning (Lai et al., 2024). This framework tackles the challenges of diverse action spaces and complex web structures, demonstrating significant improvement in real-world tasks with its AutoWebBench benchmark. MMInA evaluates agents on multihop, multimodal tasks across evolving real-world websites (Zhang et al., 2024e). The benchmark includes 1,050 tasks requiring sequential reasoning and multimodal integration to complete compositional objectives, such as comparing products across platforms. WebCanvas pioneers a dynamic evaluation framework to assess agents in live web environments (Pan et al., 2024). Its Mind2Web-Live dataset captures the adaptability of agents to interface changes and includes metrics like key-node-based intermediate evaluation, fostering progress in online web agent research.

开放世界交互环境挑战智能体在内容和界面不断变化的动态真实网站中导航。WebVLN 引入了一个用于网站视觉与语言导航的新基准，要求智能体解读视觉和文本指令以完成如回答用户查询等任务 (Chen 等，2024)。它通过整合 HTML 结构与渲染网页，强调多模态推理，为现实网页导航奠定基础。WebVoyager 利用大型语言模型 (LLM) 在 15 个真实网站上执行端到端导航，涵盖多样任务 (He 等，2024b)。其多模态方法整合截图和 HTML 内容，使其在动态在线环境中具备强健的决策能力。AutoWebGLM 通过 HTML 简化和强化学习优化网页导航 (Lai 等，2024)。该框架应对多样动作空间和复杂网页结构的挑战，在其 AutoWebBench 基准中展示了现实任务的显著提升。MMInA 评估智能体在不断演变的真实网站上执行多跳、多模态任务 (Zhang 等，2024e)。该基准包含 1050 个任务，要求顺序推理和多模态整合以完成组合目标，如跨平台产品比较。WebCanvas 开创了一个动态评估框架，用于评估智能体在实时网页环境中的表现 (Pan 等，2024)。其 Mind2Web-Live 数据集捕捉智能体对界面变化的适应能力，并包含基于关键节点的中间评估指标，促进在线网页智能体研究的进展。

Open-world environments are ideal for achieving both realism and dynamism. However, getting consistent evaluation and reproducibility is difficult as they evaluate agents on live websites that are subject to frequent changes.

开放世界环境理想地实现了现实性和动态性。然而，由于评估智能体时依赖频繁变化的实时网站，保持一致的评估和可复现性较为困难。

3.3 Evaluation Metrics

3.3 评估指标

3.3.1 Task Completion Metrics.

3.3.1 任务完成指标。

The majority of benchmarks use task completion rate as the primary metric to measure GUI agents' performance. However, different papers define task completion differently. Success can be defined as whether an agent successfully stops at a goal state (Chen et al., 2024; Zhou et al., 2023b), with Zhou et al. (2023b) programmatically checking if the intended outcome has been achieved (e.g., a comment has been posted, or a form has been completed), or whether the returned results exactly match the ground truth labels (Shi et al., 2017; Yao et al., 2022; Koh et al., 2024a; Drouin et al., 2024; Levy et al., 2024; Mialon et al., 2023). Another approach is to measure success based on whether an agent completes all required subtasks (Lai et al., 2024; Zhang et al., 2024e; Pan et al., 2024; Furuta et al., 2023; Jang et al., 2024; Cheng et al., 2024b). This approach can be further extended to measure partial success, as shown in Zhang et al. (2024e). WebVoyager uses GPT-4V to automatically determine success based on the agent's trajectory, reporting a high agreement rate of 85.3% with human judgments (He et al., 2024b). Instead of using a single final-state success metric, WebLINX measures an overall success rate based on aggregated turn-level success metrics across tasks (Lù et al., 2024). These turn-level metrics, including Intersection over Union and F1, are computed based on the type of action taken. Lastly, there are task-specific metrics to measure success, e.g., using ROUGE-L, F1 for open-ended generation (Liu et al., 2024b; Xu et al., 2024; Wornow et al., 2024), accuracy for multiple choice question tasks (Liu et al., 2024b), Precision and Recall for Standard Operating Procedure validation (Wornow et al., 2024).

大多数基准测试使用任务完成率作为衡量图形用户界面 (GUI) 智能体性能的主要指标。然而，不同论文对任务完成的定义有所不同。成功可以定义为智能体是否成功停在目标状态 (Chen 等人，2024；Zhou 等人，2023b)，其中 Zhou 等人 (2023b) 通过编程检查预期结果是否达成 (例如，是否发布了评论或完成了表单)，或者返回的结果是否与真实标签完全匹配 (Shi 等人，2017；Yao 等人，2022；Koh 等人，2024a；Drouin 等人，2024；Levy 等人，2024；Mialon 等人，2023)。另一种方法是根据智能体是否完成所有必需的子任务来衡量成功 (Lai 等人，2024；Zhang 等人，2024e；Pan 等人，2024；Furuta 等人，2023；Jang 等人，2024；Cheng 等人，2024b)。这种方法可以进一步扩展以衡量部分成功，如 Zhang 等人 (2024e) 所示。WebVoyager 使用 GPT - 4V 根据智能体的轨迹自动判断成功与否，报告显示其与人类判断的一致率高达 85.3%(He 等人，2024b)。WebLINX 不是使用单一的最终状态成功指标，而是根据跨任务聚合的回合级成功指标来衡量总体成功率 (Lù 等人，2024)。这些回合级指标，包括交并比 (Intersection over Union) 和 F1 分数，是根据所采取的行动类型计算的。最后，还有特定任务的指标来衡量成功，例如，对于开放式生成任务使用 ROUGE - L、F1 分数 (Liu 等人，2024b；Xu 等人，2024；Wornow 等人，2024)，对于多项选择题任务使用准确率 (Liu 等人，2024b)，对于标准操作程序验证使用精确率和召回率 (Wornow 等人，2024)。

3.3.2 Intermediate Step Metrics.

3.3.2 中间步骤指标

While the task completion rate is a single straightforward metric that simplifies evaluation, it fails to provide clear insights into their specific behaviors. Although some fine-grained metrics measure step-wise performance, their scope remains limited. WebCanvas (Pan et al., 2024) evaluates step scores using three distinct targets: URL Matching, which verifies whether the agent navigated to the correct webpage; Element Path Matching, which checks if the agent interacted with the appropriate UI element, such as a button or text box; and Element Value Matching, which ensures the agent inputted or extracted the correct values, such as filling a form or reading text. WebLINX (Lù et al., 2024) uses an intent match metric to assess whether the predicted action's intent aligns with the reference intent. Similarly, Mind2Web (Deng et al., 2023) and MT-Mind2Web (Deng et al., 2024) evaluate

Element Accuracy by measuring the rate at which the agent selects the correct elements. These systems also measure the precision, recall, and F1 score for token-level operations, such as clicking or typing, and calculate the Step Success Rate, which reflects the proportion of individual task steps completed correctly. While step-wise evaluations provide more fine-grained insight into the agent's performance, it is often challenging to collect reference labels at the step level while also providing enough flexibility to consider different paths to achieve the original tasks.

虽然任务完成率是一个简单直接的指标，简化了评估过程，但它无法清晰地洞察智能体的具体行为。尽管一些细粒度指标可以衡量逐步性能，但它们的范围仍然有限。WebCanvas(Pan 等人，2024) 使用三个不同的目标来评估步骤得分:URL 匹配，用于验证智能体是否导航到了正确的网页；元素路径匹配，用于检查智能体是否与合适的用户界面 (UI) 元素 (如按钮或文本框) 进行了交互；元素值匹配，用于确保智能体输入或提取了正确的值 (如填写表单或读取文本)。WebLINX(Lù 等人，2024) 使用意图匹配指标来评估预测动作的意图是否与参考意图一致。同样，Mind2Web(Deng 等人，2023) 和 MT - Mind2Web(Deng 等人，2024) 通过测量智能体选择正确元素的比率来评估元素准确率。这些系统还会测量令牌级操作 (如点击或输入) 的精确率、召回率和 F1 分数，并计算步骤成功率，该指标反映了单个任务步骤正确完成的比例。虽然逐步评估能更细致地洞察智能体的性能，但通常很难在步骤级别收集参考标签，同时还要提供足够的灵活性以考虑实现原始任务的不同路径。

3.3.3 Efficiency, Generalization, Safety and Robustness Metrics.

3.3.3 效率、泛化性、安全性和鲁棒性指标

Lastly, we summarize additional metrics that evaluate various aspects of GUI agents beyond their raw performance. Existing benchmarks include metrics for efficiency (Shahbandeh et al., 2024; Chen et al., 2024; Shahbandeh et al., 2024), generalization across diverse or compositional task settings (Furuta et al., 2023), adherence to safety policies (Levy et al., 2024), and robustness to environmental distractions (Ma et al., 2024).

最后，我们总结了一些额外的指标，这些指标用于评估图形用户界面 (GUI) 智能体除原始性能之外的各个方面。现有的基准测试包括评估效率的指标 (Shahbandeh 等人，2024；Chen 等人，2024；Shahbandeh 等人，2024)、在不同或组合任务设置下的泛化性指标 (Furuta 等人，2023)、遵守安全策略的指标 (Levy 等人，2024) 以及对环境干扰的鲁棒性指标 (Ma 等人，2024)。

4 GUI Agent Architectures

4 图形用户界面 (GUI) 智能体架构

This section focuses on various architectural designs of a GUI agent, which we categorize into four main types: (1) Perception: designs that enable the GUI agent to perceive and interpret observations from its environment; (2) Reasoning: designs related to the cognitive processes of a GUI agent, such as using an external knowledge base for long-term memory access or a world model of the environment to support other modules like planning; (3) Planning: designs related to decomposing a task into subtasks and creating a plan for their execution; and (4) Acting: mechanisms that allow the GUI agent to interact with the environment, including representing actions in

natural language using specific templates, JSON, or programming languages as action representations. We present a taxonomy of GUI agent architectures in Figure 1.

本节重点介绍 GUI 代理的各种架构设计，我们将其分为四大类:(1) 感知: 使 GUI 代理能够感知并解释其环境中的观察结果的设计；(2) 推理: 与 GUI 代理的认知过程相关的设计，例如使用外部知识库进行长期记忆访问或利用环境的世界模型支持规划等其他模块；(3) 规划: 与将任务分解为子任务并制定执行计划相关的设计；(4) 执行: 允许 GUI 代理与环境交互的机制，包括使用特定模板、JSON 或编程语言作为动作表示的自然语言动作表达。我们在图 1 中展示了 GUI 代理架构的分类体系。

4.1 Perception

4.1 感知

Unlike API-based agents that process structured, program-readable data (Xu et al., 2025b), GUI agents must perceive and understand the on-screen environment that is designed for human consumption. This requires carefully chosen interfaces that allow agents to discover the location, identity, and properties of the interactive elements. Broadly, these perception interfaces can be categorized into four types: accessibility-based, HTML/DOM-based, screen-visual-based, and hybrid ones, with each offering different capabilities and posing distinct privacy and implementation considerations.

与处理结构化、程序可读数据的基于 API 的代理 (Xu 等，2025b) 不同，GUI 代理必须感知并理解为人类设计的屏幕环境。这需要精心选择的接口，使代理能够发现交互元素的位置、身份和属性。总体而言，这些感知接口可分为四类: 基于辅助功能的、基于 HTML/DOM 的、基于屏幕视觉的和混合型，每种接口提供不同的能力，并带来不同的隐私和实现考量。

4.1.1 Accessibility-Based Interfaces

4.1.1 基于辅助功能的接口

Modern mobile and desktop operating systems usually provide accessibility APIs[2] that expose a semantic hierarchy of UI components, including their roles, labels, and states [345] . GUI agents can utilize accessibility APIs to identify actionable elements and derive semantic cues without relying solely on pixel-based detection. These interfaces are resilient to minor layout changes or styling updates; however, their effectiveness depends on proper implementation by developers. Accessibility APIs may also be limited when dealing with highly dynamic elements (e.g., custom drawing canvases or gaming environments) and may not natively expose visual content. Although these APIs help reduce the complexity of visually parsing the screen, the agent may need additional perception methods for full functionality. On the positive side, accessibility-based interfaces typically require minimal sensitive user data, thereby reducing privacy concerns.

现代移动和桌面操作系统通常提供辅助功能 APIs[2]，暴露 UI 组件的语义层级，包括其角色、标签和状态 [3][4][5]。GUI 代理可以利用辅助功能 API 识别可操作元素并获取语义线索，而不必仅依赖基于像素的检测。这些接口对轻微的布局变化或样式更新具有较强的鲁棒性；然而，其效果依赖于开发者的正确实现。辅助功能 API 在处理高度动态元素 (如自定义绘图画布或游戏环境) 时可能受限，且通常不直接暴露视觉内容。尽管这些 API 有助于降低视觉解析屏幕的复杂性，代理可能仍需额外的感知方法以实现完整功能。积极方面是，基于辅助功能的接口通常只需极少的敏感用户数据，从而减少隐私问题。

4.1.2 HTML/DOM-Based Interfaces

4.1.2 基于 HTML/DOM 的接口

For web GUIs, agents frequently utilize the Document Object Model (DOM) to interpret the structural layout of a page. The DOM provides a hierarchical representation of elements, allowing agents to locate targets like buttons or input fields based on tags, attributes, or text content. However, raw HTML data or DOM tree usually has redundant and noisy structure. Various methods are proposed to handle this. Mind2Web (Deng et al., 2023) utilizes a fine-tuned small LM to rank the elements in a page before the final prediction of action with a large LM, and WebAgent (Gur et al., 2023) uses a specialized model HTML-T5 to generate task-specific HTML snippets. AutoWebGLM (Lai et al., 2024) designs an algorithm to simplify HTML content. While HTML/DOM-based interfaces provide rich structural data, they require careful preprocessing and, in some cases, additional heuristics or trained models to locate and interpret key UI components accurately.

对于网页 GUI，代理常利用文档对象模型 (DOM) 来解析页面的结构布局。DOM 提供元素的层级表示，使代理能够基于标签、属性或文本内容定位按钮或输入字段等目标。然而，原始 HTML 数据或 DOM 树通常包含冗余和噪声结构。为此提出了多种处理方法。Mind2Web(Deng 等，2023) 利用微调的小型语言模型对页面元素进行排序，再用大型语言模型进行最终动作预测；WebAgent(Gur 等，2023) 使用专门模型 HTML-T5 生成特定任务的 HTML 片段；AutoWebGLM(Lai 等，2024) 设计了简化 HTML 内容的算法。虽然基于 HTML/DOM 的接口提供丰富的结构数据，但需要仔细预处理，并在某些情况下借助额外的启发式方法或训练模型以准确定位和解释关键 UI 组件。

4.1.3 Screen-visual-based Interfaces

4.1.3 基于屏幕视觉的接口

With advances in computer vision and multimodal LLMs, agents can utilize screen-visual information, such as screenshots, to perceive the on-screen environment. OmniParser (Lu et al., 2024) utilizes an existing multimodal LLM (e.g., GPT-4V) to parse a screenshot into a structured representation of the UI elements. TAG (Xu et al., 2025a) leverages the inherent attention patterns in pretrained MLLMs to improve GUI grounding without the need for additional fine-tuning. Cradle (Tan et al., 2024), instead of relying on a single screenshot, processes a video recording (i.e., a sequence of screenshots) to enable more general-purpose computer control. However, screen-visual-based perception introduces privacy concerns since entire screenshots may contain sensitive information. Additionally, computational overhead increases as models must handle high-dimensional image inputs. Despite these challenges, such interfaces are crucial for agents operating in environments where high-quality ac-

cessibility interfaces and DOM information are unavailable, or environments where dynamic or visual information is crucial, like image or video editing software. A key advantage of this approach is that it requires no application instrumentation, enabling direct deployment across a wide range of applications.

随着计算机视觉和多模态大型语言模型 (LLM) 的进步，代理可以利用屏幕视觉信息，如截图，来感知屏幕环境。OmniParser(Lu 等，2024) 利用现有多模态 LLM(如 GPT-4V) 将截图解析为 UI 元素的结构化表示。TAG(Xu 等，2025a) 利用预训练多模态 LLM 内在的注意力模式提升 GUI 定位，无需额外微调。Cradle(Tan 等，2024) 不依赖单张截图，而是处理视频录制 (即一系列截图)，以实现更通用的计算机控制。然而，基于屏幕视觉的感知带来隐私问题，因为完整截图可能包含敏感信息。此外，模型需处理高维图像输入，计算开销增加。尽管如此，这类接口对于在缺乏高质量辅助功能接口和 DOM 信息的环境中运行的代理，或在动态或视觉信息关键的环境 (如图像或视频编辑软件) 中尤为重要。该方法的一个关键优势是无需应用程序的额外嵌入，能够直接部署于广泛的应用中。

---

[2] https://en.wikipedia.org/wiki/Computer_accessibility

[2] https://en.wikipedia.org/wiki/Computer_accessibility

[3] https://developer.apple.com/library/archive/ documentation/Accessibility/Conceptual/ AccessibilityMacOSX/OSX-AXmodel.html

[3] https://developer.apple.com/library/archive/ documentation/Accessibility/Conceptual/ AccessibilityMacOSX/OSXAXmodel.html

[4] https://developer.apple.com/design/ human-interface-guidelines/accessibility

[4] https://developer.apple.com/design/ human-interface-guidelines/accessibility

[5] https://learn.microsoft.com/en-us/windows/apps/design/ accessibility/accessibility

[5] https://learn.microsoft.com/en-us/windows/apps/design/ accessibility/accessibility

---

4.1.4 Hybrid Interfaces

4.1.4 混合接口

To achieve robust and flexible performance across diverse environments, many GUI agents employ a hybrid approach (Gou et al., 2024; Wu et al., 2024b; Kil et al., 2024). These systems combine accessibility APIs, DOM data, and screen-visual information to form a more comprehensive understanding of the interface. Leading methods in GUI agent tasks, such as OS-Atlas (Wu et al., 2024b) and UGround (Gou et al., 2024), demonstrate that hybrid interfaces that combine visual and textual inputs can enhance performance. Such approaches also facilitate error recovery, when accessibility or DOM data are incomplete or misleading, the agent can fall back on screen parsing, and vice versa.

为了在多样化环境中实现稳健且灵活的性能，许多 GUI 代理采用混合方法 (Gou 等，2024；Wu 等，2024b；Kil 等，2024)。这些系统结合了辅助功能 API、DOM 数据和屏幕视觉信息，以形成对界面的更全面理解。GUI 代理任务中的领先方法，如 OS-Atlas(Wu 等，2024b) 和 UGround(Gou 等，2024)，表明结合视觉和文本输入的混合界面能够提升性能。这类方法还促进了错误恢复，当辅助功能或 DOM 数据不完整或误导时，代理可以依赖屏幕解析，反之亦然。

4.2 Reasoning

4.2 推理

WebPilot employs a dual optimization strategy for reasoning (Zhang et al., 2024d). WebOccam improves reasoning by refining the observation and action space of LLM agents (Yang et al., 2024). OSCAR introduces a general-purpose agent to generate Python code from human instructions (Wang and Liu, 2024). LAST leverages LLMs for reasoning, acting, and planning (Zhou et al., 2023a).

WebPilot 采用双重优化策略进行推理 (Zhang 等，2024d)。WebOccam 通过优化大型语言模型 (LLM) 代理的观察和动作空间来提升推理能力 (Yang 等，2024)。OSCAR 引入通用代理，根据人类指令生成 Python 代码 (Wang 和 Liu，2024)。LAST 利用 LLM 进行推理、执行和规划 (Zhou 等，2023a)。

4.3 Planning

4.3 规划

Planning involves decomposing a global task into multiple subtasks that progressively approach the goal state starting from an initial state (Huang et al., 2024). Traditional planning methods, such as symbolic approaches (Kautz and Selman, 1992) and reinforcement learning (Sutton and Barto, 1998), have significant limitations: symbolic methods require extensive human expertise to define rigid system rules and lack error tolerance (Belta et al., 2007; Pallagani et al., 2022), while reinforcement learning demands impractical volumes of training data, often derived from costly environmental interactions (Acharya et al., 2023). Recent advancements in LLM-powered agents offer a transformative alternative by positioning LLM-powered agents as the cognitive core for planning agents (Huang et al., 2024). When equipping agents with GUIs as the medium, LLM-powered agents can directly interact with nearly all application domains and resources to enhance planning strategies. Based on what application domains/resources agents use for planning, we divide existing works into planning with internal and external knowledge.

规划涉及将全局任务分解为多个子任务，从初始状态逐步接近目标状态 (Huang 等，2024)。传统规划方法，如符号方法 (Kautz 和 Selman，1992) 和强化学习 (Sutton 和 Barto，1998)，存在显著局限: 符号方法需要大量人类专业知识来定义严格的系统规则且缺乏容错性 (Belta 等，2007；Pallagani 等，2022)，而强化学习则需要大量训练数据，通常来自昂贵的环境交互 (Acharya 等，2023)。近期基于 LLM 的代理带来了变革性替代方案，将 LLM 代理定位为规划代理的认知核心 (Huang 等，2024)。当为代理配备 GUI 作为媒介时，LLM 代理可以直接与几乎所有应用领域和资源交互，以增强规划策略。根据代理用于规划的应用领域/资源，我们将现有工作分为基于内部知识和外部知识的规划。

4.3.1 Planning with Internal Knowledge

4.3.1 基于内部知识的规划

Planning with internal knowledge of GUI agents is to leverage the inherent knowledge to reason and think about the potential plans to fulfill the global task goals (Schraagen et al., 2000). Web-Dreamer (Gu et al., 2024) uses LLMs to simulate the outcomes of the actions of each agent and then evaluates the result to determine the optimal plan at each step. MobA (Zhu et al., 2024) devises a two-level architecture to power the mobile phone management, with a high level for understanding user commands, tracking history memories and planning tasks, and a low level to act the planned module. Agent S (Agashe et al., 2024) introduces an experience-augmented hierarchical planning to perform complex computer tasks.

基于 GUI 代理的内部知识进行规划，是利用固有知识推理和思考潜在计划以实现全局任务目标 (Schraagen 等，2000)。Web-Dreamer(Gu 等，2024) 使用 LLM 模拟每个代理动作的结果，然后评估结果以确定每步的最优计划。MobA(Zhu 等，2024) 设计了两级架构支持手机管理，高层负责理解用户命令、跟踪历史记忆和规划任务，低层执行规划模块。Agent S(Agashe 等，2024) 引入经验增强的分层规划以执行复杂计算机任务。

4.3.2 Planning with External Knowledge

4.3.2 基于外部知识的规划

Enabling LLM-powered agents to interact with diverse applications and resources through GUIs allows them to leverage external data sources, thereby enhancing their planning capabilities. For example, Search-Agent (Koh et al., 2024b) combines LLM inference with A* search to explore and backtrack to alternative paths explicitly, AgentQ (Putta et al., 2024) combines LLM with MCTS. Toolchain (Zhuang et al., 2023) models tool planning as a tree search algorithm and incorporates A* search to adaptively retrieve the most promising tool for subsequent use based on accumulated and anticipated costs. SGC (Wu et al., 2024a) decomposes the query and performs embedding similarity match between the concatenated subquery with the current retrieved task API and each of the existing APIs, and then selects the top one from the existing neighboring APIs. Thought Propagation Retrieval (Yu et al., 2023) prompts LLMs to propose a set of analogous problems and then applies established prompting techniques, like Chain-of-Thought, to derive solutions. The aggregation module subsequently consolidates solutions from these analogous problems, enhancing the problem-solving process for the original input. Benchmarks like WebShop, Mind2Web, and WebArena (Zhou et al., 2023c; Deng et al., 2023) enable agents to interact with web environments to plan and execute browsing actions for information-seeking tasks. WMA (Chae et al., 2024) utilizes world models to address the mistakes made by LLMs for long-horizon tasks.

使基于 LLM 的代理通过 GUI 与多样化应用和资源交互，使其能够利用外部数据源，从而增强规划能力。例如，Search-Agent(Koh 等，2024b) 结合 LLM 推理与 A* 搜索，显式探索并回溯备选路径；AgentQ(Putta 等，2024) 结合 LLM 与蒙特卡洛树搜索 (MCTS)。Toolchain(Zhuang 等，2023) 将工具规划建模为树搜索算法，并结合 A* 搜索，根据累计和预期成本自适应检索最有前景的工具以供后续使用。SGC(Wu 等，2024a) 将查询分解，并对拼接的子查询与当前检索的任务 API 及现有 API 进行嵌入相似度匹配，随后从邻近 API 中选取最优者。思维传播检索 (Thought Propagation Retrieval)(Yu 等，2023) 促使 LLM 提出一组类比问题，然后应用链式思维 (Chain-of-Thought) 等成熟提示技术推导解决方案。聚合模块随后整合这些类比问题的解决方案，提升原始输入的问题解决过程。诸如 WebShop、Mind2Web 和 WebArena(Zhou 等，2023c；Deng 等，2023) 等基准使代理能够与网页环境交互，规划并执行浏览操作以完成信息检索任务。WMA(Chae 等，2024) 利用世界模型解决 LLM 在长时任务中出现的错误。

4.4 Acting

4.4 执行

Acting in GUI agents involves translating the agent's reasoning and planning outputs into executable steps within the GUI environment. Unlike purely text-based or API-driven agents, GUI agents must articulate their actions at a finer granularity-often down to pixel-level coordinates-while also handling higher-level semantic actions such as typing text, scrolling, or clicking on specific elements. Several approaches have emerged:

在 GUI 代理中，执行操作涉及将代理的推理和规划输出转化为 GUI 环境中可执行的步骤。与纯文本或基于 API 的代理不同，GUI 代理必须以更细粒度表达其操作——通常精确到像素级坐标——同时还需处理诸如输入文本、滚动或点击特定元素等更高层次的语义操作。出现了几种方法:

Those utilizing textual interfaces may only rely on text-based metadata (HTML, accessibility trees) to identify UI elements. For example, WebAgent (Gur et al., 2023) and Mind2Web (Deng et al., 2023) use DOM or HTML representations to locate interactive elements. Similarly, AppAgent (Zhang et al., 2023) and MobileAgent (Wang et al., 2024a) leverage accessibility APIs to identify GUI components on mobile platforms.

那些利用文本接口的方法可能仅依赖基于文本的元数据 (HTML、辅助功能树) 来识别 UI 元素。例如，WebAgent(Gur 等，2023) 和 Mind2Web(Deng 等，2023) 使用 DOM 或 HTML 表示来定位交互元素。类似地，AppAgent(Zhang 等，2023) 和 MobileAgent(Wang 等，2024a) 利用辅助功能 API 识别移动平台上的 GUI 组件。

However, as highlighted in UGround (Gou et al., 2024), such metadata can be noisy, incomplete, and computationally expensive to parse at every step. To overcome these limitations, recent research emphasizes visual-only grounding-mapping textual referring expressions or instructions directly to pixel-level coordinates on a screenshot. UGround trains large action models using only screen-level visual inputs. OmniParser (Lu et al., 2024) also demonstrates how vision-only approaches can parse GUIs without HTML or accessibility data. Similarly, OS-Atlas (Wu et al., 2024b) leverages large-scale multi-platform training data to achieve universal GUI grounding that generalizes across web, mobile, and desktop platforms. By unifying data sources and action schemas, OS-Atlas showcases the feasibility of a universal approach to action grounding.

然而，正如 UGround(Gou 等，2024) 所指出的，这类元数据可能存在噪声、不完整，且在每一步解析时计算开销较大。为克服这些限制，近期研究强调纯视觉定位——将文本指称表达或指令直接映射到截图上的像素级坐标。UGround 仅使用屏幕级视觉输入训练大型动作模型。OmniParser(Lu 等，2024) 也展示了如何仅凭视觉方法解析 GUI，无需 HTML 或辅助功能数据。类似地，OS-Atlas(Wu 等，2024b) 利用大规模多平台训练数据，实现了跨网页、移动和桌面平台的通用 GUI 定位。通过统一数据源和动作模式，OS-Atlas 展示了通用动作定位方法的可行性。

5 GUI Agent Training Methods

5 GUI 代理训练方法

This section summarizes different strategies to elicit the ability to solve agentic tasks in a GUI Agent agent. We broadly categorize these strategies into two types: (1) Prompt-based Methods and (2) Training-based Methods. Prompt-based methods do not involve the training of parameters; they elicit the ability to solve agentic tasks by providing detailed instructions within the prompt. Training-based methods, on the other hand, involve optimizing the agent's parameters to maximize an objective, such as pretraining, fine-tuning, or reinforcement learning. We present a taxonomy of GUI agent training methods in Figure 2.

本节总结了激发 GUI 代理解决代理任务能力的不同策略。我们将这些策略大致分为两类:(1) 基于提示的方法和 (2) 基于训练的方法。基于提示的方法不涉及参数训练；它们通过在提示中提供详细指令来激发解决代理任务的能力。基于训练的方法则通过优化代理参数以最大化某一目标 (如预训练、微调或强化学习) 来实现。我们在图 2 中展示了 GUI 代理训练方法的分类。

5.1 Prompt-based Methods

5.1 基于提示的方法

Prompt-based methods enable GUI agents to exhibit learning and adaptation during inference through carefully designed prompts and interaction mechanisms, without modifying model parameters. This learning and adaptation occur as the agent's state evolves by incorporating context from past actions or stored knowledge.

基于提示的方法使 GUI 代理能够在推理过程中通过精心设计的提示和交互机制表现出学习和适应能力，而无需修改模型参数。这种学习和适应随着代理状态的演变而发生，代理通过整合过去操作的上下文或存储的知识实现。

Agent Q (Putta et al., 2024) and OSCAR (Wang and Liu, 2024) incorporate self-reflection and self-critique mechanisms via prompts, enabling agents to iteratively improve decision-making by identifying and rectifying errors. Auto-Intent (Kim et al., 2024) focuses on unsupervised intent discovery and utilization, extracting intents from interaction histories and incorporating them into future prompts. Other techniques include state-space exploration in LASER (Ma et al., 2023), state machine in OSCAR (Wang and Liu, 2024), expert development and multi-agent collaboration in MobileExperts (Zhang et al., 2024b), and app memory in AutoDroid (Wen et al., 2024).

Agent Q(Putta 等，2024) 和 OSCAR(Wang 和 Liu，2024) 通过提示引入自我反思和自我批评机制，使代理能够通过识别和纠正错误迭代改进决策。Auto-Intent(Kim 等，2024) 专注于无监督意图发现与利用，从交互历史中提取意图并将其纳入未来提示。其他技术包括 LASER(Ma 等，2023) 中的状态空间探索，OSCAR(Wang 和 Liu，2024) 中的状态机，MobileExperts(Zhang 等，2024b) 中的专家开发与多代理协作，以及 AutoDroid(Wen 等，2024) 中的应用记忆。

Despite the potential of prompt-based methods, the limited context size of LLMs and the difficulty of designing effective prompts that elicit the desired behavior remain.

尽管基于提示的方法潜力巨大，但 LLM 上下文大小有限以及设计有效提示以激发期望行为的难度依然存在。

5.2 Training-based Methods

5.2 基于训练的方法

5.2.1 Pre-training

5.2.1 预训练

Earlier models for GUI tasks relied on assembling smaller encoder-decoder architectures to address visual understanding challenges due to its ability to learn unified representations from diverse visual and textual data, enhance transfer learning capabilities, and integrate multiple modalities deeply. For example, PIX2STRUCT (Lee et al., 2023) is pre-trained on a screenshot parsing task, which involves predicting simplified HTML representations from screenshots with visually masked regions. It employs a ViT (Dosovitskiy, 2020) as the image encoder, T5 (Raffel et al., 2020) as the text encoder, and a Transformer-based decoder.

早期的 GUI 任务模型依赖组装较小的编码器-解码器架构来解决视觉理解挑战，因为其能够从多样的视觉和文本数据中学习统一表示，增强迁移学习能力，并深度整合多模态信息。例如，PIX2STRUCT(Lee 等，2023) 在截图解析任务上进行预训练，该任务涉及从带有视觉遮挡区域的截图预测简化的 HTML 表示。它采用 ViT(Dosovitskiy，2020) 作为图像编码器，T5(Raffel 等，2020) 作为文本编码器，以及基于 Transformer 的解码器。

Training of recent GUI agent models often involve the continual pre-training of existing vision large language models on additional large-scale datasets. This step refines the model's general knowledge and modifies or assembles new neural network modules into the backbone, providing a stronger foundation before fine-tuning on smaller, curated datasets for GUI tasks. VisionLLM (Wang et al., 2023) utilizes public datasets to integrate BERT (Devlin, 2018) and Deformable DETR (Zhu et al., 2020) into large language models, focusing on visual question answering tasks centered on grounding and detection. SeeClick (Cheng et al., 2024a) is built using continual pre-training on Qwen-VL (Bai et al., 2023) with datasets incorporating OCR-based layout annotation to predict click actions. UGround (Gou et al., 2024) uses continual pre-training on the LLaVA-NEXT (Liu et al., 2024a) model without its low-resolution image fusion module on a large dataset and synthetic data to align visual elements with HTML metadata for planning and grounding tasks.

近期 GUI 代理模型的训练通常涉及对现有视觉大语言模型 (Vision Large Language Models) 在额外大规模数据集上的持续预训练。此步骤细化了模型的通用知识，并将新的神经网络模块修改或组装进主干网络,为在较小的、精心挑选的 GUI 任务数据集上进行微调提供更坚实的基础。VisionLLM(Wang 等，2023) 利用公开数据集将 BERT(Devlin，2018) 和 Deformable DETR(Zhu 等，2020) 集成到大语言模型中，专注于以定位和检测为核心的视觉问答任务。SeeClick(Cheng 等，2024a) 基于 Qwen-VL(Bai 等，2023) 进行持续预训练，使用包含基于 OCR 的布局标注的数据集来预测点击动作。UGround(Gou 等，2024) 在 LLaVA-NEXT(Liu 等，2024a) 模型 (去除其低分辨率图像融合模块) 上进行持续预训练，利用大规模数据集和合成数据，将视觉元素与 HTML 元数据对齐，用于规划和定位任务。

Pre-training is also used to adapt new designs for improved computational efficiency in GUI-related tasks. CogAgent (Hong et al., 2023) employs a high-resolution cross-module to process small icons and text, enhancing its efficiency for GUI tasks such as DOM element generation and action prediction. ShowUI (Lin et al., 2024) builds on Qwen2-VL (Wang et al., 2024c) with a visual-token selection module to improve the computational efficiency for interleaved high-resolution grounding.

预训练也被用于适应新设计以提升 GUI 相关任务的计算效率。CogAgent(Hong 等，2023) 采用高分辨率跨模块处理小图标和文本，提升了其在 DOM 元素生成和动作预测等 GUI 任务中的效率。ShowUI(Lin 等，2024) 基于 Qwen2-VL(Wang 等，2024c) 构建，配备视觉标记选择模块，以提高交错高分辨率定位的计算效率。

5.2.2 Fine-tuning

5.2.2 微调

Fine-tuning has emerged as a key strategy to adapt large vision-language models (VLMs) and large language models (LLMs) to the specialized domain of GUI interaction. Unlike zero-shot or prompt-only approaches, fine-tuning can enhance both the model's grounding in GUI elements and its ability to execute instructions reliably.

微调已成为将大型视觉语言模型 (VLMs) 和大型语言模型 (LLMs) 适应 GUI 交互专业领域的关键策略。与零样本或仅提示方法不同，微调能够增强模型对 GUI 元素的定位能力及其执行指令的可靠性。

Recent work highlights reducing hallucinations and improving grounding. Falcon-UI (Shen et al., 2024a) fine-tunes on large-scale instruction-free GUI data and then fine-tunes on Android and Web tasks, achieving high accuracy with fewer parameters. VGA (Ziyang et al., 2024), through image-centric fine-tuning, reduces hallucinations by tightly coupling visual inputs with GUI elements, thus improving action reliability. Similarly, UI-Pro (Li et al., 2024) identifies a recipe for fine-tuning of VLMs, reducing model size while maintaining state-of-the-art grounding accuracy.

近期研究强调减少幻觉现象和提升定位准确性。Falcon-UI(Shen 等，2024a) 在大规模无指令 GUI 数据上进行微调，随后在 Android 和 Web 任务上微调，以较少参数实现高准确率。VGA(Ziyang 等，2024) 通过以图像为中心的微调，将视觉输入与 GUI 元素紧密结合，减少幻觉现象，从而提升动作的可靠性。同样，UI-Pro(Li 等，2024) 提出了 VLM 微调方案，缩小模型规模的同时保持最先进的定位准确率。

Other methods leverage fine-tuning to incorporate domain-specific reasoning and functionalities, such as functionality-aware fine-tuning for generating human-like interactions (Liu et al., 2024d) and alignment strategies to handle multilingual, variable-resolution GUI inputs (Nong et al., 2024). Some methods emphasize autonomous adaptation, such as learning to execute arbitrary voice commands through trial-and-error exploration (Pan et al., 2023) and learning for cross-platform GUI grounding without structured text (Cheng et al., 2024a). Additionally, fine-tuning can specialize models for context-sensitive actions. Techniques proposed by Liu et al. (2023) enable context-aware text input generation, improving coverage in GUI testing scenarios. Taken together, these fine-tuning methods demonstrate how careful parameter adaptation, data scaling and multimodal alignment can collectively advance the reliability, interpretability, and performance of GUI agents.

其他方法利用微调引入领域特定的推理和功能，如面向功能的微调以生成类人交互 (Liu 等，2024d) 以及处理多语言、多分辨率 GUI 输入的对齐策略 (Nong 等，2024)。部分方法强调自主适应能力，如通过试错探索学习执行任意语音命令 (Pan 等，2023) 和无结构文本的跨平台 GUI 定位学习 (Cheng 等，2024a)。此外，微调还能使模型专注于上下文敏感的动作。Liu 等 (2023) 提出的技术支持上下文感知的文本输入生成，提升 GUI 测试场景的覆盖率。综上，这些微调方法展示了通过精细的参数调整、数据扩展和多模态对齐，如何共同推动 GUI 代理的可靠性、可解释性和性能提升。

5.2.3 Reinforcement Learning

5.2.3 强化学习

Reinforcement learning was used in the early text-based agent WebGPT to improve information retrieval of the GPT-3 based model (Nakano et al., 2021). Liu et al. (2018) use human demonstrations to constrain the search space for RL, through using workflows as a high-level process for the model to complete without specifying the specific details. An example from Liu et al. (2018) is for the specific process of forwarding a given email, the workflow would involve clicking forward, typing in the address, and clicking send. Deng et al. (2023) use RL based on human demonstrations as the reward signal. While early agents constrained the input and action spaces to only text, recent work has extended to GUI agents.

强化学习曾被用于早期基于文本的代理 WebGPT，以提升基于 GPT-3 模型的信息检索能力 (Nakano 等，2021)。Liu 等 (2018) 通过人类示范约束强化学习的搜索空间，利用工作流作为模型完成的高级流程，而不指定具体细节。Liu 等 (2018) 举例说明了转发特定邮件的流程，工作流包括点击转发、输入地址和点击发送。Deng 等 (2023) 基于人类示范将强化学习作为奖励信号。早期代理将输入和动作空间限制为文本，近期研究已扩展至 GUI 代理。

WebRL framework uses RL to generate new tasks based on previously unsuccessful attempts as a mitigation for sparse rewards (Qi et al., 2024). Task success is evaluated by an LLM-based outcome reward model (ORM) and KL-divergence is used to prevent significant shifts in policies during curriculum learning. AutoGLM applies online, curriculum learning, in particular to address error recovery during real-world use and to correct for stochasticity not present in simulators (Liu et al., 2024c). DigiRL uses a modified advantage-weighted regression (AWR) algorithm for offline learning (Peng et al., 2019), but modifies AWR for more stochastic environments by using a simple value function and curriculum learning.

WebRL 框架使用强化学习 (RL) 基于先前未成功的尝试生成新任务，以缓解稀疏奖励问题 (Qi 等，2024)。任务成功通过基于大型语言模型 (LLM) 的结果奖励模型 (ORM) 进行评估，且在课程学习过程中使用 KL 散度防止策略发生显著偏移。AutoGLM 采用在线课程学习，特别用于解决现实世界使用中的错误恢复问题，并纠正模拟器中不存在的随机性 (Liu 等，2024c)。DigiRL 使用修改后的优势加权回归 (AWR) 算法进行离线学习 (Peng 等，2019)，但通过使用简单的价值函数和课程学习对 AWR 进行了调整，以适应更具随机性的环境。

6 Open Problems & Challenges

6 开放问题与挑战

User Intent Understanding. GUI Agents still struggle to accurately infer user goals across diverse applications, achieving only 51.1% accuracy on unseen websites (Kim et al., 2024). Designing models that generalize effectively across varying tasks is crucial, particularly for handling contextual variations in user interactions (Stefanidi et al., 2022) and predicting user behavior in complex interfaces (Gao et al., 2024). A prospective future research direction is to leverage robust training techniques to enable agents to adapt to new environments with minimal retraining, ultimately providing more seamless and adaptive user experiences. Other promising directions could include training on diverse user interaction datasets and incorporating context-aware learning techniques that utilize historical user actions to better predict intent.

用户意图理解。GUI 代理在不同应用中仍难以准确推断用户目标，在未见过的网站上仅达到 51.1% 的准确率 (Kim 等，2024)。设计能够有效泛化至多样任务的模型至关重要，尤其是在处理用户交互中的上下文变化 (Stefanidi 等，2022) 和预测复杂界面中的用户行为 (Gao 等，2024) 方面。未来研究方向之一是利用鲁棒训练技术，使代理能够以最少的再训练适应新环境，最终提供更流畅和自适应的用户体验。其他有前景的方向包括在多样化的用户交互数据集上训练，以及结合利用历史用户行为的上下文感知学习技术以更好地预测意图。

Security and Privacy. GUI agents frequently interact with sensitive data such as passwords, confidential documents, and personal credentials, raising serious privacy and security concerns (He et al., 2024a; Zhang et al., 2024a). These risks are further amplified when agents rely on cloud-based processing, which involves transmitting sensitive information to remote servers. Unauthorized access or incorrect actions could result in severe consequences (Zhang et al., 2024c). Future research could focus on developing privacy-preserving protocols, such as homomorphic encryption or differential privacy, to ensure data remains secure during both inference and storage. Additional directions may include exploring local processing alternatives and implementing advanced authentication mechanisms to enhance the reliability and safety of GUI agents across diverse environments.

安全与隐私。GUI 代理经常处理密码、机密文件和个人凭证等敏感数据，带来严重的隐私和安全隐患 (He 等，2024a；Zhang 等，2024a)。当代理依赖云端处理时，这些风险进一步加剧，因为敏感信息需传输至远程服务器。未经授权的访问或错误操作可能导致严重后果 (Zhang 等，2024c)。未来研究可聚焦于开发隐私保护协议，如同态加密或差分隐私，确保数据在推理和存储过程中的安全。其他方向包括探索本地处理方案和实施先进的认证机制，以提升 GUI 代理在多样环境中的可靠性和安全性。

Inference Latency. The need to manage complex interactions across diverse applications often conflicts with

the requirement for real-time responsiveness. Optimizing model efficiency without compromising accuracy remains a key challenge, particularly when deploying agents in resource-constrained environments. Key issues include minimizing computational overhead, leveraging hardware acceleration, and balancing trade-offs between speed and resource usage. Addressing these challenges calls for lightweight model architectures and adaptive techniques that enable timely, seamless interactions in dynamic GUI settings. Future research could investigate hardware-aware optimization methods, such as quantization and pruning, or efficient decoding strategies like predictive sampling and multi-token prediction, which can significantly reduce latency while preserving system accuracy.

推理延迟。管理跨多样应用的复杂交互需求常与实时响应要求相冲突。优化模型效率而不牺牲准确性仍是关键挑战，尤其是在资源受限环境中部署代理时。主要问题包括最小化计算开销、利用硬件加速，以及在速度与资源使用之间权衡。解决这些挑战需要轻量级模型架构和自适应技术，以实现动态 GUI 环境中的及时无缝交互。未来研究可探讨硬件感知的优化方法，如量化和剪枝，或高效解码策略，如预测采样和多标记预测，这些方法能显著降低延迟同时保持系统准确性。

Personalization. is a pivotal aspect in the development of GUI agents, aiming to tailor interactions to individual user preferences and behaviors, thereby enhancing satisfaction and efficiency. Recent work (Berkovitch et al., 2024) introduced a method for identifying user goals from UI trajectories, enabling agents to infer intentions and proactively assist users based on their interactions with the interface. Future research could explore more sophisticated models that incorporate user feedback to refine personalization strategies, while ensuring trust and compliance with data protection regulations. Additional directions include implementing explicit feedback mechanisms (e.g., thumbs-up/thumbs-down ratings) and developing robust user profiling techniques that integrate behavioral and contextual data to enable more meaningful and adaptive personalization.

个性化。是 GUI 代理开发中的关键方面，旨在根据个体用户偏好和行为定制交互，从而提升满意度和效率。近期工作 (Berkovitch 等，2024) 提出了一种从 UI 轨迹识别用户目标的方法，使代理能够基于用户与界面的交互推断意图并主动协助用户。未来研究可探索更复杂的模型，结合用户反馈以优化个性化策略，同时确保信任和遵守数据保护法规。其他方向包括实现显式反馈机制 (如点赞/点踩评分) 和开发融合行为与上下文数据的稳健用户画像技术，以实现更有意义和自适应的个性化。

7 Conclusion

7 结论

In this survey, we have thoroughly explored GUI Agents, examining various benchmarks, agent architectures, and training methods. Although considerable strides have been made, problems such as intent understanding, security, latency, and personalization remain critical challenges. We hope that this survey is a valuable resource for researchers, offering structure and practical guidance in this rapidly growing and exciting field, and inspiring more work on GUI Agents. The progress in this area has already benefited mankind, enhancing our daily productivity and transforming the way we interact with computers.

在本综述中，我们全面探讨了 GUI 代理，审视了各种基准、代理架构和训练方法。尽管取得了显著进展，但意图理解、安全性、延迟和个性化等问题仍是关键挑战。我们希望本综述能为研究人员提供结构化和实用的指导，助力这一快速发展且令人振奋的领域，并激发更多关于 GUI 代理的研究工作。该领域的进展已惠及人类，提升了我们的日常生产力，改变了我们与计算机的交互方式。

Limitations

限制

We recognize that some studies have explored interactions between LFM-based agents and digital systems through interfaces other than GUIs, such as Command Line Interfaces (CLI) (Nguyen et al., 2024) or Application Programming Interfaces (API). (Song et al., 2025) However, these approaches are relatively limited in scope compared to GUI-based methods. To maintain a focused scope for our survey, we have chosen not to include them in our discussion.

我们认识到，一些研究已经通过图形用户界面 (GUI) 以外的接口，如命令行界面 (CLI)(Nguyen 等人，2024) 或应用程序编程接口 (API)(Song 等人，2025)，探索了基于大型语言模型 (LFM) 的智能体与数字系统之间的交互。然而，与基于 GUI 的方法相比，这些方法的应用范围相对有限。为了使我们的综述重点突出，我们决定不在讨论中涉及这些方法。

References

参考文献

Kamal Acharya, Waleed Raza, Carlos Dourado, Alvaro Velasquez, and Houbing Herbert Song. 2023. Neurosymbolic reinforcement learning and planning: A survey. IEEE Transactions on Artificial Intelligence.

卡迈勒·阿查里亚 (Kamal Acharya)、瓦利德·拉扎 (Waleed Raza)、卡洛斯·多拉多 (Carlos Dourado)、阿尔瓦罗·贝拉斯克斯 (Alvaro Velasquez) 和宋鹤兵 (Houbing Herbert Song)。2023 年。神经符号强化学习与规划: 综述。《IEEE 人工智能汇刊》。

Saaket Agashe, Jiuzhou Han, Shuyu Gan, Jiachen Yang, Ang Li, and Xin Eric Wang. 2024. Agent s: An open agentic framework that uses computers like a human.

萨基特·阿加什 (Saaket Agashe)、韩九州 (Jiuzhou Han)、甘舒雨 (Shuyu Gan)、杨嘉晨 (Jiachen Yang)、李昂 (Ang Li) 和王昕 (Xin Eric Wang)。2024 年。智能体 (Agents): 一个像人类一样使用计算机的开放智能体框架。

Hao Bai, Yifei Zhou, Mert Cemri, Jiayi Pan, Alane Suhr, Sergey Levine, and Aviral Kumar. 2024. Di-girl: Training in-the-wild device-control agents with autonomous reinforcement learning.

白浩 (Hao Bai)、周逸飞 (Yifei Zhou)、梅尔特·杰姆里 (Mert Cemri)、潘佳宜 (Jiayi Pan)、阿莱恩·苏尔 (Alane Suhr)、谢尔盖·莱文 (Sergey Levine) 和阿维拉·库马尔 (Aviral Kumar)。2024 年。Di - girl: 通过自主强化学习训练野外设备控制智能体。

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. arXiv preprint arXiv:2308.12966.

白晋泽 (Jinze Bai)、白帅 (Shuai Bai)、杨树生 (Shusheng Yang)、王世杰 (Shijie Wang)、谭思楠 (Sinan Tan)、王鹏 (Peng Wang)、林俊阳 (Junyang Lin)、周长 (Chang Zhou) 和周靖人 (Jingren Zhou)。2023 年。Qwen - vl: 一种具有多种能力的前沿大型视觉语言模型。预印本 arXiv:2308.12966。

Calin Belta, Antonio Bicchi, Magnus Egerstedt, Emilio Frazzoli, Eric Klavins, and George J Pappas. 2007. Symbolic planning and control of robot motion [grand challenges of robotics]. IEEE Robotics & Automation Magazine, 14(1):61-70.

卡林·贝尔塔 (Calin Belta)、安东尼奥·比基 (Antonio Bicchi)、马格努斯·埃格斯泰德 (Magnus Egerstedt)、埃米利奥·弗拉佐利 (Emilio Frazzoli)、埃里克·克拉文斯 (Eric Klavins) 和乔治·J·帕帕斯 (George J Pappas)。2007 年。机器人运动的符号规划与控制 [机器人学的重大挑战]。《IEEE 机器人与自动化杂志》，14(1):61 - 70。

Omri Berkovitch, Sapir Caduri, Noam Kahlon, Anatoly Efros, Avi Caciularu, and Ido Dagan. 2024. Identifying user goals from ui trajectories. ArXiv preprint, abs/2406.14314.

奥姆里·贝科维奇 (Omri Berkovitch)、萨皮尔·卡杜里 (Sapir Caduri)、诺姆·卡隆 (Noam Kahlon)、阿纳托利·埃弗罗斯 (Anatoly Efros)、阿维·卡丘拉鲁 (Avi Caciularu) 和伊多·达甘 (Ido Dagan)。2024 年。从用户界面轨迹中识别用户目标。预印本，abs/2406.14314。

Léo Boisvert, Megh Thakkar, Maxime Gasse, Massimo Caccia, Thibault Le Sellier De Chezelles, Quentin Cappart, Nicolas Chapados, Alexandre Lacoste, and Alexandre Drouin. 2024. Workarena++: Towards compositional planning and reasoning-based common knowledge work tasks. In Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024.

莱奥·博伊斯韦特 (Léo Boisvert)、梅格·萨克 (Megh Thakkar)、马克西姆·加斯 (Maxime Gasse)、马西莫·卡恰 (Massimo Caccia)、蒂博·勒·塞利耶·德·谢泽勒 (Thibault Le Sellier De Chezelles)、昆汀·卡帕尔 (Quentin Cappart)、尼古拉斯·查帕多斯 (Nicolas Chapados)、亚历山大·拉科斯特 (Alexandre Lacoste) 和亚历山大·德鲁安 (Alexandre Drouin)。2024 年。Workarena++: 迈向基于组合规划和推理的常识性工作任务。《神经信息处理系统进展 38:2024 年神经信息处理系统年度会议，NeurIPS 2024，加拿大不列颠哥伦比亚省温哥华，2024 年 12 月 10 - 15 日》。

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori

Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Lad-hak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchan-dani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Pa-padimitriou, Joon Sung Park, Chris Piech, Eva Porte-lance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the opportunities and risks of foundation models.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Lad-hak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchan-dani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Pa-padimitriou, Joon Sung Park, Chris Piech, Eva Porte-lance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. 关于基础模型 (foundation models) 的机遇与风险。

Rogerio Bonatti, Dan Zhao, Francesco Bonacci, Dillon Dupont, Sara Abdali, Yinheng Li, Yadong Lu, Justin Wagle, Kazuhito Koishida, Arthur Bucker, Lawrence Jang, and Zack Hui. 2024. Windows agent arena: Evaluating multi-modal os agents at scale.

Rogerio Bonatti, Dan Zhao, Francesco Bonacci, Dillon Dupont, Sara Abdali, Yinheng Li, Yadong Lu, Justin Wagle, Kazuhito Koishida, Arthur Bucker, Lawrence Jang, 和 Zack Hui. 2024. Windows 代理竞技场: 大规模评估多模态操作系统代理。

Ruisheng Cao, Fangyu Lei, Haoyuan Wu, Jixuan Chen, Yeqiao Fu, Hongcheng Gao, Xinzhuang Xiong, Hanchong Zhang, Wenjing Hu, Yuchen Mao, Tianbao Xie, Hongshen Xu, Danyang Zhang, Sida I. Wang, Ruoxi Sun, Pengcheng Yin, Caiming Xiong, Ansong Ni, Qian Liu, Victor Zhong, Lu Chen, Kai Yu, and Tao Yu. 2024. Spider2-v: How far are multimodal agents from automating data science and engineering workflows? In Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024.

Ruisheng Cao, Fangyu Lei, Haoyuan Wu, Jixuan Chen, Yeqiao Fu, Hongcheng Gao, Xinzhuang Xiong, Hanchong Zhang, Wenjing Hu, Yuchen Mao, Tianbao Xie, Hongshen Xu, Danyang Zhang, Sida I. Wang, Ruoxi Sun, Pengcheng Yin, Caiming Xiong, Ansong Ni, Qian Liu, Victor Zhong, Lu Chen, Kai Yu, 和 Tao Yu. 2024. Spider2-v: 多模态代理距离自动化数据科学与工程工作流还有多远？发表于《神经信息处理系统进展》第 38 届年会 (NeurIPS 2024)，加拿大温哥华，2024 年 12 月 10-15 日。

Hyungjoo Chae, Namyoung Kim, Kai Tzu iunn Ong, Minju Gwak, Gwanwoo Song, Jihoon Kim, Sungh-wan Kim, Dongha Lee, and Jinyoung Yeo. 2024. Web agents with world models: Learning and leveraging environment dynamics in web navigation.

Hyungjoo Chae, Namyoung Kim, Kai Tzu iunn Ong, Minju Gwak, Gwanwoo Song, Jihoon Kim, Sunghwan Kim, Dongha Lee, 和 Jinyoung Yeo. 2024. 具备世界模型的网络代理: 学习并利用环境动态进行网页导航。

Qi Chen, Dileepa Pitawela, Chongyang Zhao, Gengze Zhou, Hsiang-Ting Chen, and Qi Wu. 2024. Web-vln: Vision-and-language navigation on websites. In Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada, pages 1165- 1173. AAAI Press.

Qi Chen, Dileepa Pitawela, Chongyang Zhao, Gengze Zhou, Hsiang-Ting Chen, 和 Qi Wu. 2024. Web-vln: 基于视觉与语言的网站导航。发表于第三十八届美国人工智能协会会议 (AAAI 2024)、第三十六届创新人工智能应用会议 (IAAI 2024) 及第十四届人工智能教育进展研讨会 (EAAI 2014)，2024 年 2 月 20-27 日，加拿大温哥华，页码 1165-1173。AAAI 出版社。

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024a. Seeclick: Harnessing gui grounding for advanced visual gui agents. ArXiv preprint, abs/2401.10935.

程坤志，孙求是，褚有刚，徐方志，李艳涛，张建兵，吴志勇。2024a。Seeclick: 利用 GUI 定位实现高级视觉 GUI 代理。ArXiv 预印本，abs/2401.10935。

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024b. Seeclick: Harnessing gui grounding for advanced visual gui agents.

程坤志，孙求是，褚有刚，徐方志，李艳涛，张建兵，吴志勇。2024b。Seeclick: 利用 GUI 定位实现高级视觉 GUI 代理。

Yuheng Cheng, Ceyao Zhang, Zhengwen Zhang, Xi-angrui Meng, Sirui Hong, Wenhao Li, Zihao Wang, Zekai Wang, Feng Yin, Junhua Zhao, and Xiuqiang He. 2024c. Exploring large language model based intelligent agents: Definitions, methods, and prospects.

程宇恒，张策尧，张正文，孟西昂锐，洪思睿，李文浩，王子豪，王泽凯，尹峰，赵俊华，何修强。2024c。探索基于大型语言模型 (LLM) 的智能代理: 定义、方法与前景。

Thibault Le Sellier De Chezelles, Maxime Gasse, Alexandre Drouin, Massimo Caccia, Léo Boisvert, Megh Thakkar, Tom Marty, Rim Assouel, Sa-har Omidi Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, De-

han Kong, Frank F. Xu, Siva Reddy, Quentin Cappart, Graham Neubig, Rus-lan Salakhutdinov, Nicolas Chapados, and Alexandre Lacoste. 2024. The browsergym ecosystem for web agent research. CoRR, abs/2412.05467.

Thibault Le Sellier De Chezelles，Maxime Gasse，Alexandre Drouin，Massimo Caccia，Léo Boisvert，Megh Thakkar，Tom Marty，Rim Assouel，Sa-har Omidi Shayegan，Lawrence Keunho Jang，Xing Han Lù，Ori Yoran，Dehan Kong，Frank F. Xu，Siva Reddy，Quentin Cappart，Graham Neubig，Rus-lan Salakhutdinov，Nicolas Chapados，Alexandre Lacoste。2024。BrowserGym 生态系统: 面向网页代理研究。CoRR，abs/2412.05467。

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

姜伟林，李卓翰，林子，盛颖，吴章浩，张浩，郑连民，庄思远，庄永浩，Joseph E. Gonzalez，Ion Stoica，Eric P. Xing。2023。Vicuna: 一款开源聊天机器人，达到 90%* ChatGPT 质量，令 GPT-4 印象深刻。

Sumit Kumar Dam, Choong Seon Hong, Yu Qiao, and Chaoning Zhang. 2024. A complete survey on llm-based ai chatbots.

Sumit Kumar Dam，Choong Seon Hong，乔宇，张朝宁。2024。基于大型语言模型 (LLM) 的 AI 聊天机器人的完整综述。

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samual Stevens, Boshi Wang, Huan Sun, and Yu Su. 2023. Mind2web: Towards a generalist agent for the web. In Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023.

邓翔，顾宇，郑博远，陈世杰，Samuel Stevens，王博时，孙欢，苏宇。2023。Mind2Web: 迈向通用网页代理。在《神经信息处理系统进展》第 36 届会议论文集，NeurIPS 2023，美国新奥尔良，2023 年 12 月 10-16 日。

Yang Deng, Xuan Zhang, Wenxuan Zhang, Yifei Yuan, See-Kiong Ng, and Tat-Seng Chua. 2024. On the multi-turn instruction following for conversational web agents.

邓洋，张轩，张文轩，袁一飞，吴思强，蔡达生。2024。关于对话式网页代理的多轮指令跟随研究。

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Jacob Devlin。2018。BERT: 用于语言理解的深度双向变换器预训练。arXiv 预印本, arXiv:1810.04805。

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

Alexey Dosovitskiy。2020。一张图像胜过 16x16 个词: 大规模图像识别的变换器。arXiv 预印本，arXiv:2010.11929。

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Is-sam H. Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. 2024. Workarena: How capable are web agents at solving common knowledge work tasks?

Alexandre Drouin，Maxime Gasse，Massimo Caccia，Issam H. Laradji，Manuel Del Verme，Tom Marty，Léo Boisvert，Megh Thakkar，Quentin Cappart，David Vazquez，Nicolas Chapados，Alexandre Lacoste。2024。WorkArena: 网页代理解决常识性工作任务的能力如何?

Hiroki Furuta, Yutaka Matsuo, Aleksandra Faust, and Izzeddin Gur. 2023. Language model agents suffer from compositional generalization in web automation. ArXiv preprint, abs/2311.18751.

古田浩树，松尾丰，Aleksandra Faust，Izzeddin Gur。2023。语言模型代理在网页自动化中存在组合泛化问题。ArXiv 预印本，abs/2311.18751。

Difei Gao, Lei Ji, Zechen Bai, Mingyu Ouyang, Peiran Li, Dongxing Mao, Qinchen Wu, Weichen Zhang, Peiyi Wang, Xiangwu Guo, et al. 2024. Assistgui: Task-oriented pc graphical user interface automation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 13289- 13298.

高迪飞，纪磊，白泽辰，欧阳明宇，李佩然，毛东兴，吴秦辰，张伟晨，王佩怡，郭祥武，等。2024。AssistGUI: 面向任务的 PC 图形用户界面自动化。发表于 IEEE/CVF 计算机视觉与模式识别会议论文集，页码 13289-13298。

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. 2024. Navigating the digital world as humans do: Universal visual grounding for gui agents.

苟博宇，王若涵，郑博远，谢雅楠，常成，舒一恒，孙欢，苏宇。2024。像人类一样导航数字世界: 面向 GUI 代理的通用视觉定位。

Yu Gu, Boyuan Zheng, Boyu Gou, Kai Zhang, Cheng Chang, Sanjari Srivastava, Yanan Xie, Peng Qi, Huan Sun, and Yu Su. 2024. Is your llm secretly a world model of the internet? model-based planning for web agents.

顾瑜，郑博远，苟博宇，张凯，常成，Sanjari Srivastava，谢亚楠，齐鹏，孙焕，苏瑜。2024。你的大型语言模型 (LLM) 是否暗中成为了互联网的世界模型? 基于模型的网页代理规划。

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

Izzeddin Gur，古田浩树，Austin Huang，Mustafa Safdari，松尾丰，Douglas Eck，Aleksandra Faust。2023。具备规划、长上下文理解和程序合成能力的真实世界网页代理。

Milos Hauskrecht. 2000. Value-function approximations for partially observable Markov decision processes. Journal of Artificial Intelligence Research, 13:33-94.

Milos Hauskrecht。2000。部分可观测马尔可夫决策过程 (POMDP) 的价值函数近似。《人工智能研究杂志》，13:33-94。

Feng He, Tianqing Zhu, Dayong Ye, Bo Liu, Wanlei Zhou, and Philip S Yu. 2024a. The emerged security and privacy of llm agent: A survey with case studies. ArXiv preprint, abs/2407.19354.

何峰，朱天庆，叶大勇，刘波，周万磊，Philip S Yu。2024a。大型语言模型代理的安全与隐私新兴问题: 带案例研究的综述。ArXiv 预印本，abs/2407.19354。

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. 2024b. Webvoyager: Building an end-to-end web agent with large multimodal models.

何洪亮，姚文林，马凯鑫，余文浩，戴勇，张洪明，兰振中，余东。2024b。Webvoyager: 基于大型多模态模型构建端到端网页代理。

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxuan Zhang, Juanzi Li, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. 2023. Cogagent: A visual language model for gui agents.

洪文怡，王伟涵，吕庆松，徐家政，余文萌，季俊辉，王岩，王子涵，张宇轩，李娟子，徐斌，董宇霄，丁明，唐杰。2023。Cogagent: 面向图形用户界面代理的视觉语言模型。

Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruim-ing Tang, and Enhong Chen. 2024. Understanding the planning of llm agents: A survey. ArXiv preprint, abs/2402.02716.

黄旭，刘伟文，陈晓龙，王杏梅，王浩，连德福，王亚胜，唐瑞明，陈恩宏。2024。理解大型语言模型代理的规划: 综述。ArXiv 预印本，abs/2402.02716。

Iat Long Iong, Xiao Liu, Yuxuan Chen, Hanyu Lai, Shuntian Yao, Pengbo Shen, Hao Yu, Yuxiao Dong, and Jie Tang. 2024. Openwebagent: An open toolkit to enable web agents on large language models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 72-81.

梁逸龙，刘晓，陈宇轩，赖涵宇，姚顺天，沈鹏博，余浩，董宇霄，唐杰。2024。Openwebagent: 支持大型语言模型网页代理的开源工具包。载于第 62 届计算语言学协会年会论文集 (第 3 卷: 系统演示)，第 72-81 页。

Lawrence Jang, Yinheng Li, Charles Ding, Justin Lin, Paul Pu Liang, Dan Zhao, Rogerio Bonatti, and Kazuhito Koishida. 2024. Videowebarena: Evaluating long context multimodal agents with video understanding web tasks.

Lawrence Jang，李银恒，丁查尔斯，林贾斯汀，梁浦，赵丹，Rogerio Bonatti，小石田一仁。2024。Videowebarena: 基于视频理解网页任务评估长上下文多模态代理。

Sayash Kapoor, Rishi Bommasani, Kevin Klyman, Shayne Longpre, Ashwin Ramaswami, Peter Ci-hon, Aspen Hopkins, Kevin Bankston, Stella Bider-man, Miranda Bogen, Rumman Chowdhury, Alex Engler, Peter Henderson, Yacine Jernite, Seth Lazar, Stefano Maffulli, Alondra Nelson, Joelle Pineau, Aviya Skowron, Dawn Song,

Victor Storchan, Daniel Zhang, Daniel E. Ho, Percy Liang, and Arvind Narayanan. 2024. On the societal impact of open foundation models.

Sayash Kapoor，Rishi Bommasani，Kevin Klyman，Shayne Longpre，Ashwin Ramaswami，Peter Cihon，Aspen Hopkins，Kevin Bankston，Stella Biderman，Miranda Bogen，Rumman Chowdhury，Alex Engler，Peter Henderson，Yacine Jernite，Seth Lazar，Stefano Maffulli，Alondra Nelson，Joelle Pineau，Aviya Skowron，Dawn Song，Victor Storchan，Daniel Zhang，Daniel E. Ho，Percy Liang，Arvind Narayanan。2024。开放基础模型的社会影响。

Henry A. Kautz and Bart Selman. 1992. Planning as satisfiability. In 10th European Conference on Artificial Intelligence, ECAI 92, Vienna, Austria, August 3-7, 1992. Proceedings, pages 359-363. John Wiley and Sons.

Henry A. Kautz，Bart Selman。1992。规划作为可满足性问题。载于第 10 届欧洲人工智能会议 (ECAI 92)，奥地利维也纳，1992 年 8 月 3-7 日，论文集，第 359-363 页。John Wiley and Sons 出版。

Jihyung Kil, Chan Hee Song, Boyuan Zheng, Xiang Deng, Yu Su, and Wei-Lun Chao. 2024. Dual-view visual contextualization for web navigation. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024, pages 14445-14454. IEEE.

Jihyung Kil, 宋灿熙, 郑博远, 邓翔, 苏瑜, 赵伟伦。2024。网页导航的双视角视觉语境化。载于 IEEE/CVF 计算机视觉与模式识别会议 (CVPR 2024)，美国西雅图，2024 年 6 月 16-22 日，第 14445-14454 页。IEEE 出版。

Jaekyeom Kim, Dong-Ki Kim, Lajanugen Logeswaran, Sungryull Sohn, and Honglak Lee. 2024. Auto-intent: Automated intent discovery and self-exploration for large language model web agents. ArXiv preprint, abs/2410.22552.

Jaekyeom Kim，Dong-Ki Kim，Lajanugen Logeswaran，Sungryull Sohn，Honglak Lee。2024。Auto-intent: 大型语言模型网页代理的自动意图发现与自我探索。ArXiv 预印本，abs/2410.22552。

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. 2024a. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks. ArXiv preprint, abs/2401.13649.

Jing Yu Koh，Robert Lo，Lawrence Jang，Vikram Duvvur，Ming Chong Lim，Po-Yu Huang，Graham Neubig，Shuyan Zhou，Ruslan Salakhutdinov，Daniel Fried。2024a。Visualwebarena: 评估多模态代理在真实视觉网页任务中的表现。ArXiv 预印本，abs/2401.13649。

Jing Yu Koh, Stephen McAleer, Daniel Fried, and Rus-lan Salakhutdinov. 2024b. Tree search for language model agents.

许景瑜 (Jing Yu Koh)、斯蒂芬·麦勒 (Stephen McAleer)、丹尼尔·弗里德 (Daniel Fried) 和鲁斯兰·萨拉胡季诺夫 (Ruslan Salakhutdinov)。2024b。语言模型智能体的树搜索。

Hanyu Lai, Xiao Liu, Iat Long Iong, Shuntian Yao, Yux-uan Chen, Pengbo Shen, Hao Yu, Hanchen Zhang, Xiaohan Zhang, Yuxiao Dong, and Jie Tang. 2024. Autowebglm: A large language model-based web navigating agent.

赖瀚宇 (Hanyu Lai)、刘晓 (Xiao Liu)、严逸朗 (Iat Long Iong)、姚舜添 (Shuntian Yao)、陈宇轩 (Yuxuan Chen)、申鹏博 (Pengbo Shen)、余浩 (Hao Yu)、张瀚晨 (Hanchen Zhang)、张晓晗 (Xiaohan Zhang)、董宇霄 (Yuxiao Dong) 和唐杰 (Jie Tang)。2024。Autowebglm: 一种基于大语言模型的网页导航智能体。

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexi-ang Hu, Fangyu Liu, Julian Martin Eisenschlos, Ur-vashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2023. Pix2struct: Screen-shot parsing as pretraining for visual language understanding. In International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, volume 202 of Proceedings of Machine Learning Research, pages 18893-18912. PMLR.

肯顿·李 (Kenton Lee)、曼达尔·乔希 (Mandar Joshi)、尤利娅·拉卢卡·图尔克 (Iulia Raluca Turc)、胡鹤翔 (Hexiang Hu)、刘方宇 (Fangyu Liu)、朱利安·马丁·艾森施洛斯 (Julian Martin Eisenschlos)、乌尔瓦希·坎德尔瓦尔 (Urvashi Khandelwal)、彼得·肖 (Peter Shaw)、张明伟 (Ming-Wei Chang) 和克里斯蒂娜·图托纳娃 (Kristina Toutanova)。2023。Pix2struct: 将屏幕截图解析作为视觉语言理解的预训练。收录于《2023 年国际机器学习会议 (ICML 2023) 论文集》，2023 年 7 月 23 - 29 日，美国夏威夷檀香山，《机器学习研究会议录》第 202 卷，第 18893 - 18912 页。机器学习研究会议录 (PMLR)。

Ido Levy, Ben Wiesel, Sami Marreed, Alon Oved, Avi Yaeli, and Segev Shlomov. 2024. St-webagentbench: A benchmark for evaluating safety and trustworthiness in web agents.

伊多·利维 (Ido Levy)、本·维塞尔 (Ben Wiesel)、萨米·马雷德 (Sami Marreed)、阿隆·奥韦德 (Alon Oved)、阿维·亚埃利 (Avi Yaeli) 和塞格夫·施洛莫夫 (Segev Shlomov)。2024。St-webagentbench: 一个评估网页智能体安全性和可信度的基准。

Hongxin Li, Jingran Su, Jingfan CHEN, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. 2024. UI-pro: A hidden recipe for building vision-language models for GUI grounding.

李宏鑫 (Hongxin Li)、苏静然 (Jingran Su)、陈景凡 (Jingfan CHEN)、陈云涛 (Yuntao Chen)、李清 (Qing Li) 和张兆翔 (Zhaoxiang Zhang)。2024。UI-pro: 构建用于图形用户界面 (GUI) 基础的视觉 - 语言模型的隐藏秘诀。

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Weixian Lei, Lijuan Wang, and Mike Zheng Shou. 2024. Showui: One vision-language-action model for gui visual agent.

林庆鸿 (Kevin Qinghong Lin)、李临杰 (Linjie Li)、高迪飞 (Difei Gao)、杨正远 (Zhengyuan Yang)、吴世伟 (Shiwei Wu)、白泽晨 (Zechen Bai)、雷伟贤 (Weixian Lei)、王丽娟 (Lijuan Wang) 和郑守 (Mike Zheng Shou)。2024。Showui: 一种用于图形用户界面视觉智能体的视觉 - 语言 - 动作模型。

Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tian-lin Shi, and Percy Liang. 2018. Reinforcement learning on web interfaces using workflow-guided exploration.

刘哲然 (Evan Zheran Liu)、凯尔文·顾 (Kelvin Guu)、帕努蓬·帕苏帕特 (Panupong Pasupat)、史天霖 (Tianlin Shi) 和梁珀西 (Percy Liang)。2018。使用工作流引导探索在网页界面上进行强化学习。

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

刘浩天 (Haotian Liu)、李春元 (Chunyuan Li)、李雨衡 (Yuheng Li)、李博 (Bo Li)、张元瀚 (Yuanhan Zhang)、沈盛 (Sheng Shen) 和李永宰 (Yong Jae Lee)。2024a。Llava-next: 改进的推理、光学字符识别 (OCR) 和世界知识。

Junpeng Liu, Yifan Song, Bill Yuchen Lin, Wai Lam, Graham Neubig, Yuanzhi Li, and Xiang Yue. 2024b. Visualwebbench: How far have multimodal llms evolved in web page understanding and grounding? ArXiv preprint, abs/2404.05955.

刘俊鹏 (Junpeng Liu)、宋一帆 (Yifan Song)、林雨晨 (Bill Yuchen Lin)、林伟 (Wai Lam)、格雷厄姆·纽比格 (Graham Neubig)、李远志 (Yuanzhi Li) 和岳翔 (Xiang Yue)。2024b。Visualwebbench: 多模态大语言模型在网页理解和基础方面发展到了什么程度？预印本，arXiv:2404.05955。

Xiao Liu, Bo Qin, Dongzhu Liang, Guang Dong, Hanyu Lai, Hanchen Zhang, Hanlin Zhao, Iat Long Iong, Jiadai Sun, Jiaqi Wang, et al. 2024c. Autoglm: Autonomous foundation agents for guis. ArXiv preprint, abs/2411.00820.

刘晓 (Xiao Liu)、秦博 (Bo Qin)、梁东柱 (Dongzhu Liang)、董光 (Guang Dong)、赖瀚宇 (Hanyu Lai)、张瀚晨 (Hanchen Zhang)、赵翰林 (Hanlin Zhao)、严逸朗 (Iat Long Iong)、孙佳岱 (Jiadai Sun)、王佳琪 (Jiaqi Wang) 等。2024c。Autoglm: 用于图形用户界面的自主基础智能体。预印本，arXiv:2411.00820。

Zhe Liu, Chunyang Chen, Junjie Wang, Xing Che, Yuekai Huang, Jun Hu, and Qing Wang. 2023. Fill in the blank: Context-aware automated text input generation for mobile gui testing. In 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE), pages 1355-1367. IEEE.

刘哲 (Zhe Liu)、陈春阳 (Chunyang Chen)、王俊杰 (Junjie Wang)、车星 (Xing Che)、黄跃凯 (Yuekai Huang)、胡军 (Jun Hu) 和王青 (Qing Wang)。2023。填空: 用于移动图形用户界面测试的上下文感知自动文本输入生成。收录于《2023 年电气与电子工程师协会/美国计算机协会第 45 届软件工程国际会议 (ICSE) 论文集》，第 1355 - 1367 页。电气与电子工程师协会 (IEEE)。

Zhe Liu, Chunyang Chen, Junjie Wang, Mengzhuo Chen, Boyu Wu, Xing Che, Dandan Wang, and Qing Wang. 2024d. Make llm a testing expert: Bringing human-like interaction to mobile gui testing via functionality-aware decisions. In Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, pages 1-13.

刘哲 (Zhe Liu)、陈春阳 (Chunyang Chen)、王俊杰 (Junjie Wang)、陈梦卓 (Mengzhuo Chen)、吴博宇 (Boyu Wu)、车星 (Xing Che)、王丹丹 (Dandan Wang) 和王青 (Qing Wang)。2024d。让大语言模型成为测试专家: 通过功能感知决策为移动图形用户界面测试带来类人交互。收录于《电气与电子工程师协会/美国计算机协会第 46 届软件工程国际会议论文集》，第 1 - 13 页。

Yadong Lu, Jianwei Yang, Yelong Shen, and Ahmed Awadallah. 2024. Omniparser for pure vision based gui agent. ArXiv preprint, abs/2408.00203.

卢亚东 (Yadong Lu)、杨建伟 (Jianwei Yang)、沈业龙 (Yelong Shen) 和艾哈迈德·阿瓦达拉 (Ahmed Awadallah)。2024。基于纯视觉的图形用户界面智能体的全解析器。预印本，arXiv:2408.00203。

Xing Han Lù, Zdeněk Kasner, and Siva Reddy. 2024. Weblinx: Real-world website navigation with multiturn dialogue.

陆星翰 (Xing Han Lù)、兹德涅克·卡斯纳 (Zdeněk Kasner) 和西瓦·雷迪 (Siva Reddy)。2024。Weblinx: 通过多轮对话进行真实世界的网站导航。

Kaixin Ma, Hongming Zhang, Hongwei Wang, Xiao-man Pan, Wenhao Yu, and Dong Yu. 2023. Laser: Llm agent with state-space exploration for web navigation.

马开心 (Kaixin Ma)、张宏明 (Hongming Zhang)、王宏伟 (Hongwei Wang)、潘晓曼 (Xiaoman Pan)、余文浩 (Wenhao Yu) 和于东 (Dong Yu)。2023。Laser: 具有状态空间探索能力的用于网页导航的大语言模型智能体。

Xinbei Ma, Yiting Wang, Yao Yao, Tongxin Yuan, Aston Zhang, Zhuosheng Zhang, and Hai Zhao. 2024. Caution for the environment: Multimodal agents are susceptible to environmental distractions.

马欣贝 (Xinbei Ma)、王艺婷 (Yiting Wang)、姚瑶 (Yao Yao)、袁童心 (Tongxin Yuan)、张 Aston(Aston Zhang)、张卓生 (Zhuosheng Zhang) 和赵海 (Hai Zhao)。2024。警惕环境: 多模态智能体易受环境干扰。

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. Gaia: a benchmark for general ai assistants.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun 和 Thomas Scialom. 2023. Gaia: 通用人工智能助手的基准测试。

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback, 2021. URL https://arxiv.org/abs/2112.09332.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders 等. 2021. WebGPT: 基于浏览器的人类反馈辅助问答，2021。网址 https://arxiv.org/abs/2112.09332。

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes 和 Ajmal Mian. 2023. 大型语言模型的全面综述。

Dang Nguyen, Viet Dac Lai, Seunghyun Yoon, Ryan A. Rossi, Handong Zhao, Ruiyi Zhang, Puneet Mathur, Nedim Lipka, Yu Wang, Trung Bui, Franck Dernon-court, and Tianyi Zhou. 2024. Dynasaur: Large language agents beyond predefined actions. ArXiv preprint, abs/2411.01747.

Dang Nguyen, Viet Dac Lai, Seunghyun Yoon, Ryan A. Rossi, Handong Zhao, Ruiyi Zhang, Puneet Mathur, Nedim Lipka, Yu Wang, Trung Bui, Franck Dernon-court 和 Tianyi Zhou. 2024. Dynasaur: 超越预定义动作的大型语言代理。ArXiv 预印本，abs/2411.01747。

Songqin Nong, Jiali Zhu, Rui Wu, Jiongchao Jin, Shuo Shan, Xiutian Huang, and Wenhao Xu. 2024. Mobileflow: A multimodal llm for mobile gui agent. ArXiv preprint, abs/2407.04346.

Songqin Nong, Jiali Zhu, Rui Wu, Jiongchao Jin, Shuo Shan, Xiutian Huang 和 Wenhao Xu. 2024. Mobileflow: 面向移动 GUI 代理的多模态大型语言模型。ArXiv 预印本，abs/2407.04346。

Vishal Pallagani, Bharath Muppasani, Keerthiram Mu-rugesan, Francesca Rossi, Lior Horesh, Biplav Srivastava, Francesco Fabiano, and Andrea Loreggia. 2022. Plansformer: Generating symbolic plans using transformers. ArXiv preprint, abs/2212.08681.

Vishal Pallagani, Bharath Muppasani, Keerthiram Murugesan, Francesca Rossi, Lior Horesh, Biplav Srivastava, Francesco Fabiano 和 Andrea Loreggia. 2022. Plansformer: 使用 Transformer 生成符号计划。ArXiv 预印本，abs/2212.08681。

Lihang Pan, Bowen Wang, Chun Yu, Yuxuan Chen, Xiangyu Zhang, and Yuanchun Shi. 2023. Auto-task: Executing arbitrary voice commands by exploring and learning from mobile gui. ArXiv preprint, abs/2312.16062.

Lihang Pan, Bowen Wang, Chun Yu, Yuxuan Chen, Xiangyu Zhang 和 Yuanchun Shi. 2023. Auto-task: 通过探索和学习移动 GUI 执行任意语音命令。ArXiv 预印本，abs/2312.16062。

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

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

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. 2019. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning.

Xue Bin Peng, Aviral Kumar, Grace Zhang 和 Sergey Levine. 2019. 优势加权回归: 简单且可扩展的离策略强化学习。

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. 2024. Agent q: Advanced reasoning and learning for autonomous ai agents.

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg 和 Rafael Rafailov. 2024. Agent Q: 自主人工智能代理的高级推理与学习。

Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao, Tianjie Zhang, Wei Xu, Jie Tang, and Yuxiao Dong. 2024. Webrl: Training llm web agents via self-evolving online curriculum reinforcement learning.

Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao, Tianjie Zhang, Wei Xu, Jie Tang 和 Yuxiao Dong. 2024. WebRL: 通过自我进化的在线课程强化学习训练大型语言模型网页代理。

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research, 21(140):1-67.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li 和 Peter J Liu. 2020. 探索统一文本到文本 Transformer 的迁移学习极限。《机器学习研究杂志》，21(140):1-67。

Johannes Schneider, Christian Meske, and Pauline Kuss. 2024. Foundation models: a new paradigm for artificial intelligence. Business & Information Systems Engineering, pages 1-11.

Johannes Schneider, Christian Meske 和 Pauline Kuss. 2024. 基础模型: 人工智能的新范式。《商业与信息系统工程》，第 1-11 页。

Jan Maarten Schraagen, Susan F Chipman, and Valerie L Shalin. 2000. Cognitive task analysis. Psychology Press.

Jan Maarten Schraagen, Susan F Chipman 和 Valerie L Shalin. 2000. 认知任务分析。Psychology Press 出版社。

Mobina Shahbandeh, Parsa Alian, Noor Nashid, and Ali Mesbah. 2024. Naviqate: Functionality-guided web application navigation. ArXiv preprint, abs/2409.10741.

Mobina Shahbandeh, Parsa Alian, Noor Nashid 和 Ali Mesbah. 2024. Naviqate: 基于功能引导的网页应用导航。ArXiv 预印本，abs/2409.10741。

Huawen Shen, Chang Liu, Gengluo Li, Xinlong Wang, Yu Zhou, Can Ma, and Xiangyang Ji. 2024a. Falcon-ui: Understanding gui before following user instructions. ArXiv preprint, abs/2412.09362.

Huawen Shen, Chang Liu, Gengluo Li, Xinlong Wang, Yu Zhou, Can Ma 和 Xiangyang Ji. 2024a. Falcon-ui: 在执行用户指令前理解 GUI。ArXiv 预印本，abs/2412.09362。

Yongliang Shen, Kaitao Song, Xu Tan, Wenqi Zhang, Kan Ren, Siyu Yuan, Weiming Lu, Dongsheng Li, and Yueting Zhuang. 2024b. Taskbench: Benchmarking large language models for task automation.

沈永亮，宋凯涛，谭旭，张文琦，任侃，袁思宇，陆伟明，李东升，庄越婷。2024b。Taskbench: 大型语言模型任务自动化的基准测试。

Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, and Percy Liang. 2017. World of bits: An open-domain platform for web-based agents. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, volume 70 of Proceedings of Machine Learning Research, pages 3135-3144. PMLR.

史天林，Andrej Karpathy，范林曦，Jonathan Hernandez，梁珀西。2017。World of bits: 一个面向网络代理的开放域平台。载于第 34 届国际机器学习大会 (ICML 2017)，澳大利亚悉尼，2017 年 8 月 6-11 日，机器学习研究论文集第 70 卷，页 3135-3144。PMLR。

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning.

Noah Shinn，Federico Cassano，Edward Berman，Ashwin Gopinath，Karthik Narasimhan，姚顺宇。2023。Reflexion: 具备语言强化学习能力的语言代理。

Edward Sondik. 1971. The Optimal Control of Partially Observable Markov Decision Processes. Ph.D. thesis, Stanford University.

Edward Sondik。1971。部分可观测马尔可夫决策过程 (Partially Observable Markov Decision Processes, POMDP) 的最优控制。斯坦福大学博士论文。

Yueqi Song, Frank Xu, Shuyan Zhou, and Graham Neu-big. 2025. Beyond browsing: Api-based web agents.

宋越琦，徐弗兰克，周淑妍，Graham Neu-big。2025。超越浏览: 基于 API 的网络代理。

Zinovia Stefanidi, George Margetis, Stavroula Ntoa, and George Papagiannakis. 2022. Real-time adaptation of context-aware intelligent user interfaces, for enhanced situational awareness. IEEE Access, 10:23367-23393.

Zinovia Stefanidi，George Margetis，Stavroula Ntoa，George Papagiannakis。2022。上下文感知智能用户界面的实时自适应，以增强情境感知。IEEE Access，10:23367-23393。

Richard Sutton and Andrew Barto. 1998. Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA.

Richard Sutton，Andrew Barto。1998。强化学习导论。麻省理工学院出版社，剑桥，马萨诸塞州。

Weihao Tan, Wentao Zhang, Xinrun Xu, Haochong Xia, Ziluo Ding, Boyu Li, Bohan Zhou, Junpeng Yue, Jiechuan Jiang, Yewen Li, Ruyi An, Molei Qin, Chuqiao Zong, Longtao Zheng, Yujie Wu, Xiaoqiang Chai, Yifei Bi, Tianbao Xie, Pengjie Gu, Xiyun Li, Ceyao Zhang, Long Tian, Chaojie Wang, Xinrun Wang, Börje F. Karlsson, Bo An, Shuicheng Yan, and Zongqing Lu. 2024. Cradle: Empowering foundation agents towards general computer control.

谭伟豪，张文涛，徐新润，夏浩冲，丁子洛，李博宇，周博涵，岳俊鹏，蒋杰川，李业文，安如意，秦墨磊，宗楚乔，郑龙涛，吴宇杰，柴晓强，毕一飞，谢天宝，顾鹏杰，李希云，张策尧，田龙，王超杰，王新润，Börje F. Karlsson，安波，严水成，陆宗庆。2024。Cradle: 赋能基础代理实现通用计算机控制。

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Moly-bog, Yixin Nie, Andrew Poulton, Jeremy Reizen-stein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subrama-nian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Hugo Touvron，Louis Martin，Kevin Stone，Peter Albert，Amjad Almahairi，Yasmine Babaei，Nikolay Bashlykov，Soumya Batra，Prajjwal Bhargava，Shruti Bhosale，Dan Bikel，Lukas Blecher，Cristian Canton Ferrer，Moya Chen，Guillem Cucurull，David Esiobu，Jude Fernandes，Jeremy Fu，Wenyin Fu，Brian Fuller，Cynthia Gao，Vedanuj Goswami，Naman Goyal，Anthony Hartshorn，Saghar Hosseini，Rui Hou，Hakan Inan，Marcin Kardas，Viktor Kerkez，Madian Khabsa，Isabel Kloumann，Artem Korenev，Punit Singh Koura，Marie-Anne Lachaux，Thibaut Lavril，Jenya Lee，Diana Liskovich，吕英海，毛宇宁，Xavier Martinet，Todor Mihaylov，Pushkar Mishra，Igor Moly-bog，聂一心，Andrew Poulton，Jeremy Reizen-stein，Rashi Rungta，Kalyan Saladi，Alan Schelten，Ruan Silva，Eric Michael Smith，Ranjan Subramanian，谭晓青，唐斌，Ross Taylor，Adina Williams，管建祥，徐璞新，严铮，Iliyan Zarov，张宇晨，Angela Fan，Melanie Kambadur，Sharan Narang，Aurelien Rodriguez，Robert Stojnic，Sergey Edunov，Thomas Scialom。2023。Llama 2: 开放基础及微调聊天模型。

Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. 2024a. Mobile-agent: Autonomous multi-modal mobile device agent with visual perception.

王俊阳，徐海洋，叶嘉博，闫明，沈伟舟，张骥，黄飞，桑继涛。2024a。Mobile-agent: 具备视觉感知的自主多模态移动设备代理。

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. 2024b. A survey on large language model based autonomous agents. Frontiers of Computer Science, 18(6).

王磊，马晨，冯雪阳，张泽宇，杨浩，张景森，陈志远，唐嘉凯，陈旭，林彦凯，赵文新，魏哲伟，温继荣。2024b。基于大型语言模型的自主代理综述。计算机科学前沿，18(6)。

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhi-hao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024c. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhi-hao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge 等人. 2024c. Qwen2-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。

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, and Jifeng Dai. 2023. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. In Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023.

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao 和 Jifeng Dai. 2023. Visionllm: 大型语言模型也是面向视觉任务的开放式解码器。发表于《神经信息处理系统进展》第 36 届年会，NeurIPS 2023，美国新奥尔良，2023 年 12 月 10-16 日。

Xiaoqiang Wang and Bang Liu. 2024. Oscar: Operating system control via state-aware reasoning and re-planning. ArXiv preprint, abs/2410.18963.

Xiaoqiang Wang 和 Bang Liu. 2024. Oscar: 通过状态感知推理与重新规划实现操作系统控制。ArXiv 预印本，abs/2410.18963。

Zichong Wang, Zhibo Chu, Thang Viet Doan, Shiwen Ni, Min Yang, and Wenbin Zhang. 2024d. History, development, and principles of large language models-an introductory survey.

Zichong Wang, Zhibo Chu, Thang Viet Doan, Shiwen Ni, Min Yang 和 Wenbin Zhang. 2024d. 大型语言模型的历史、发展与原理——入门综述。

Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, and Yunxin Liu. 2024. Autodroid: Llm-powered task automation in android. In Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, pages 543-557.

Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang 和 Yunxin Liu. 2024. Autodroid: 基于大型语言模型的安卓任务自动化。发表于第 30 届国际移动计算与网络年会论文集，页 543-557。

Michael Wornow, Avanika Narayan, Ben Viggiano, Ishan S Khare, Tathagat Verma, Tibor Thompson, Miguel Angel Fuentes Hernandez, Sudharsan Sundar, Chloe Trujillo, Krrish Chawla, et al. 2024. Do multimodal foundation models understand enterprise workflows? a benchmark for business process management tasks. ArXiv preprint, abs/2406.13264.

Michael Wornow, Avanika Narayan, Ben Viggiano, Ishan S Khare, Tathagat Verma, Tibor Thompson, Miguel Angel Fuentes Hernandez, Sudharsan Sundar, Chloe Trujillo, Krrish Chawla 等人. 2024. 多模态基础模型能理解企业工作流程吗? 面向业务流程管理任务的基准测试。ArXiv 预印本，abs/2406.13264。

Xixi Wu, Yifei Shen, Caihua Shan, Kaitao Song, Si-wei Wang, Bohang Zhang, Jiarui Feng, Hong Cheng, Wei Chen, Yun Xiong, et al. 2024a. Can graph learning improve task planning? ArXiv preprint, abs/2405.19119.

Xixi Wu, Yifei Shen, Caihua Shan, Kaitao Song, Si-wei Wang, Bohang Zhang, Jiarui Feng, Hong Cheng, Wei Chen, Yun Xiong 等人. 2024a. 图学习能提升任务规划吗？ArXiv 预印本，abs/2405.19119。

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, et al. 2024b. Os-atlas: A foundation action model for generalist gui agents. ArXiv preprint, abs/2410.23218.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang 等人. 2024b. Os-atlas: 面向通用图形用户界面代理的基础动作模型。ArXiv 预印本，abs/2410.23218。

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. 2024. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhou-jun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong 和 Tao Yu. 2024. Osworld: 面向真实计算机环境中开放式任务的多模态代理基准测试。

Hai-Ming Xu, Qi Chen, Lei Wang, and Lingqiao Liu. 2025a. Attention-driven GUI grounding: Leveraging pretrained multimodal large language models without fine-tuning. In AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA, pages 8851-8859. AAAI Press.

Hai-Ming Xu, Qi Chen, Lei Wang 和 Lingqiao Liu. 2025a. 基于注意力的 GUI 定位: 利用预训练多模态大型语言模型无需微调。发表于 AAAI-25，由人工智能促进协会主办，2025 年 2 月 25 日至 3 月 4 日，美国费城，页 8851-8859。AAAI 出版社。

Kevin Xu, Yeganeh Kordi, Tanay Nayak, Ado Asija, Yizhong Wang, Kate Sanders, Adam Byerly, Jingyu Zhang, Benjamin Van Durme, and Daniel Khashabi. 2024. Tur [k] ingbench: A challenge benchmark for web agents. ArXiv preprint, abs/2403.11905.

Kevin Xu, Yeganeh Kordi, Tanay Nayak, Ado Asija, Yizhong Wang, Kate Sanders, Adam Byerly, Jingyu Zhang, Benjamin Van Durme 和 Daniel Khashabi. 2024. Tur [k] ingbench: 面向网络代理的挑战基准。ArXiv 预印本，abs/2403.11905。

Nancy Xu, Sam Masling, Michael Du, Giovanni Campagna, Larry Heck, James Landay, and Monica Lam. 2021. Grounding open-domain instructions to automate web support tasks. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1022-1032, Online. Association for Computational Linguistics.

Nancy Xu, Sam Masling, Michael Du, Giovanni Campagna, Larry Heck, James Landay, 和 Monica Lam. 2021. 将开放域指令落地以自动化网页支持任务。载于 2021 年北美计算语言学协会人类语言技术分会会议论文集，页码 1022-1032，线上。计算语言学协会。

Paiheng Xu, Gang Wu, Xiang Chen, Tong Yu, Chang Xiao, Franck Dernoncourt, Tianyi Zhou, Wei Ai, and Viswanathan Swaminathan. 2025b. Skill discovery for software scripting automation via offline simulations with llms. arXiv preprint arXiv:2504.20406.

Paiheng Xu, Gang Wu, Xiang Chen, Tong Yu, Chang Xiao, Franck Dernoncourt, Tianyi Zhou, Wei Ai, 和 Viswanathan Swaminathan. 2025b. 通过大型语言模型 (LLMs) 离线模拟实现软件脚本自动化的技能发现。arXiv 预印本 arXiv:2504.20406。

Ke Yang, Yao Liu, Sapana Chaudhary, Rasool Fakoor, Pratik Chaudhari, George Karypis, and Huzefa Rangwala. 2024. Agentoccam: A simple yet strong baseline for llm-based web agents.

Ke Yang, Yao Liu, Sapana Chaudhary, Rasool Fakoor, Pratik Chaudhari, George Karypis, 和 Huzefa Rangwala. 2024. Agentoccam: 基于大型语言模型的网页代理的简单而强大的基线。

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2022. Webshop: Towards scalable real-

world web interaction with grounded language agents. In Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022.

Shunyu Yao, Howard Chen, John Yang, 和 Karthik Narasimhan. 2022. Webshop: 迈向具备落地语言代理的可扩展真实网页交互。载于神经信息处理系统第 35 届年会 (NeurIPS 2022)，新奥尔良，美国，2022 年 11 月 28 日至 12 月 9 日。

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, 和 Yuan Cao. 2023. React: 在语言模型中协同推理与行动。

Junchi Yu, Ran He, and Rex Ying. 2023. Thought propagation: An analogical approach to complex reasoning with large language models. ArXiv preprint, abs/2310.03965.

Junchi Yu, Ran He, 和 Rex Ying. 2023. 思维传播: 一种基于类比的大型语言模型复杂推理方法。ArXiv 预印本，abs/2310.03965。

Chaoyun Zhang, Shilin He, Jiaxu Qian, Bowen Li, Liqun Li, Si Qin, Yu Kang, Minghua Ma, Qingwei Lin, Saravan Rajmohan, et al. 2024a. Large language model-brained gui agents: A survey. ArXiv preprint, abs/2411.18279.

Chaoyun Zhang, Shilin He, Jiaxu Qian, Bowen Li, Liqun Li, Si Qin, Yu Kang, Minghua Ma, Qingwei Lin, Saravan Rajmohan 等. 2024a. 大型语言模型驱动的图形用户界面代理: 综述。ArXiv 预印本，abs/2411.18279。

Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. 2023. Appagent: Multimodal agents as smartphone users.

Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, 和 Gang Yu. 2023. Appagent: 作为智能手机用户的多模态代理。

Jiayi Zhang, Chuang Zhao, Yihan Zhao, Zhaoyang Yu, Ming He, and Jianping Fan. 2024b. Mobileexperts: A dynamic tool-enabled agent team in mobile devices.

Jiayi Zhang, Chuang Zhao, Yihan Zhao, Zhaoyang Yu, Ming He, 和 Jianping Fan. 2024b. Mobileexperts: 移动设备中的动态工具驱动代理团队。

Xinyu Zhang, Huiyu Xu, Zhongjie Ba, Zhibo Wang, Yuan Hong, Jian Liu, Zhan Qin, and Kui Ren. 2024c. Privacyasst: Safeguarding user privacy in tool-using large language model agents. IEEE Transactions on Dependable and Secure Computing.

Xinyu Zhang, Huiyu Xu, Zhongjie Ba, Zhibo Wang, Yuan Hong, Jian Liu, Zhan Qin, 和 Kui Ren. 2024c. Privacyasst: 保障使用工具的大型语言模型代理中的用户隐私。IEEE 可靠与安全计算汇刊。

Yao Zhang, Zijian Ma, Yunpu Ma, Zhen Han, Yu Wu, and Volker Tresp. 2024d. Webpilot: A versatile and autonomous multi-agent system for web task execution with strategic exploration. ArXiv preprint, abs/2408.15978.

Yao Zhang, Zijian Ma, Yunpu Ma, Zhen Han, Yu Wu, 和 Volker Tresp. 2024d. Webpilot: 具备战略探索能力的多功能自主多代理网页任务执行系统。ArXiv 预印本，abs/2408.15978。

Ziniu Zhang, Shulin Tian, Liangyu Chen, and Ziwei Liu. 2024e. Mmina: Benchmarking multihop multimodal internet agents.

Ziniu Zhang, Shulin Tian, Liangyu Chen, 和 Ziwei Liu. 2024e. Mmina: 多跳多模态互联网代理基准测试。

Pengyu Zhao, Zijian Jin, and Ning Cheng. 2023. An in-depth survey of large language model-based artificial intelligence agents.

Pengyu Zhao, Zijian Jin, 和 Ning Cheng. 2023. 基于大型语言模型的人工智能代理深入综述。

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. 2023a. Language agent tree search unifies reasoning acting and planning in language models. ArXiv preprint, abs/2310.04406.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, 和 Yu-Xiong Wang. 2023a. 语言代理树搜索: 统一语言模型中的推理、行动与规划。ArXiv 预印本，abs/2310.04406。

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. 2023b. Webarena: A realistic web environment for building autonomous agents.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, 和 Graham Neubig. 2023b. Webarena: 构建自主代理的真实网页环境。

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

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried 等. 2023c. We-barena: 构建自主代理的真实网页环境。ArXiv 预印本，abs/2307.13854。

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2020. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159.

朱锡州 (Xizhou Zhu)、苏伟杰 (Weijie Su)、卢乐伟 (Lewei Lu)、李斌 (Bin Li)、王小刚 (Xiaogang Wang) 和戴季峰 (Jifeng Dai)。2020 年。可变形 DETR: 用于端到端目标检测的可变形 Transformer。预印本 arXiv:2010.04159。

Zichen Zhu, Hao Tang, Yansi Li, Kunyao Lan, Yix-uan Jiang, Hao Zhou, Yixiao Wang, Situo Zhang, Liangtai Sun, Lu Chen, and Kai Yu. 2024. Moba: A two-level agent system for efficient mobile task automation.

朱梓晨 (Zichen Zhu)、唐浩 (Hao Tang)、李艳思 (Yansi Li)、兰坤瑶 (Kunyao Lan)、江逸轩 (Yixuan Jiang)、周浩 (Hao Zhou)、王一笑 (Yixiao Wang)、张思拓 (Situo Zhang)、孙良泰 (Liangtai Sun)、陈璐 (Lu Chen) 和于凯 (Kai Yu)。2024 年。Moba: 一种用于高效移动任务自动化的两级代理系统。

Yuchen Zhuang, Xiang Chen, Tong Yu, Saayan Mitra, Victor Bursztyn, Ryan A. Rossi, Somdeb Sarkhel, and Chao Zhang. 2023. Toolchain*: Efficient action space navigation in large language models with a* search.

庄雨晨 (Yuchen Zhuang)、陈翔 (Xiang Chen)、余童 (Tong Yu)、萨扬·米特拉 (Saayan Mitra)、维克多·布尔兹廷 (Victor Bursztyn)、瑞安·A·罗西 (Ryan A. Rossi)、索姆德布·萨凯尔 (Somdeb Sarkhel) 和张超 (Chao Zhang)。2023 年。Toolchain*: 使用 A* 搜索在大语言模型中进行高效动作空间导航。

Meng Ziyang, Yu Dai, Zezheng Gong, Shaoxiong Guo, Minglong Tang, and Tongquan Wei. 2024. Vga: Vision gui assistant-minimizing hallucinations through image-centric fine-tuning. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages 1261-1279.

孟紫阳、戴宇、龚泽政、郭少雄、唐明龙和魏同权。2024 年。Vga: 视觉图形用户界面助手——通过以图像为中心的微调减少幻觉。收录于《计算语言学协会研究成果:2024 年自然语言处理经验方法会议》，第 1261 - 1279 页。

Benchmark	Domain	Type	World	Highlights
RUSS (Xu et al., 2021)	Web	Dataset	Closed	Map instructions to a DSL for precise web execution
Mind2Web (Deng et al., 2023)	Web	Dataset	Closed	2000 diverse single-turn tasks
MT-Mind2Web (Deng et al., 2024)	Web	Dataset	Closed	Conversational, multi-turn variant of Mind2Web
TURKINGBENCH (Xu et al., 2024)	Crowdsourcing	Dataset	Closed	Micro-tasks, complex multimodal layouts
VisualWebBench (Liu et al., 2024b)	Web	Dataset	Closed	OCR, element grounding, action prediction
ScreenSpot (Cheng et al., 2024b)	Screenshots	Dataset	Closed	Click / type grounding direct from images
WONDERBREAD (Wornow et al., 2024)	BPM tasks	Dataset	Closed	Workflow documentation & improvement
EnvDistraction (Ma et al., 2024)	Synthetic GUI	Dataset	Closed	Measures robustness to ter/distractions
NaviQAte (Shahbandeh et al., 2024)	Web apps	Dataset	Closed	QA-framed navigation; functionality-guided
GAIA (Mialon et al., 2023)	General	Dataset	Open	Open-word multi-modal QA
WebLINX (Lù et al., 2024)	Live web	Dataset	Open	Multi-turn dialogue navigation
MiniWoB (Shi et al., 2017)	Synthetic web	Env.	Closed	Low-level mouse/keyboard skills
CompWoB (Furuta et al., 2023)	Synthetic web	Env.	Closed	Compositional, multi-step workflows
WebShop (Yao et al., 2022)	E-commerce	Env.	Closed	Shopping with instruction following
WebArena (Zhou et al., 2023b)	Self-hosted web	Env.	Closed	Long-horizon, multi-domain tasks
VisualWebArena (Koh et al., 2024a)	Self-hosted web	Env.	Closed	Adds pixel-level multimodality
WorkArena (Drouin et al., 2024)	Web, ServiceNow	Env.	Closed	Enterprise knowledge-work UIs
WorkArena++ (Boisvert et al., 2024)	Web, ServiceNow	Env.	Closed	WorkArena with harder tasks
BrowserGym (Chezelles et al., 2024)	Web	Env.	Closed	Unified gym environment consists of other web agent benchmarks
ST-WebAgentBench (Levy et al., 2024)	Self-hosted web	Env.	Closed	Safety trustworthiness metrics
VideoWebArena (Jang et al., 2024)	Video + Web	Env.	Closed	Long-context multimodal reasoning
OSWorld (Xie et al., 2024)	Windows GUI	Env.	Closed	Desktop OS interactions
WindowsAgentArena (Bonatti et al., 2024)	Windows GUI	Env.	Closed	Benchmarks cross-app Windows tasks
WebVLN (Chen et al., 2024)	Live web	Env.	Open	Vision-language navigation
WebVoyager (He et al., 2024b)	15 live sites	Env.	Open	End-to-end nav; HTML + screenshots
AutoWebBench (Lai et al., 2024)	Live web	Env.	Open	RL finetuning, HTML simplification
MMInA (Zhang et al., 2024e)	Live web	Env.	Open	Multihop, multimodal objectives
WebCanvas (Pan et al., 2024)	Live web	Env.	Open	Dynamic eval; interface-change resilience

基准测试	领域	类型	世界	亮点
RUSS(Xu 等人，2021 年)	网络	数据集	封闭的	将指令映射到特定领域语言 (DSL) 以进行精确的网络执行
Mind2Web(Deng 等人，2023 年)	网络	数据集	封闭的	2000 个不同的单轮任务
MT - Mind2Web(Deng 等人，2024 年)	网络	数据集	封闭的	Mind2Web 的对话式多轮变体
TURKINGBENCH(Xu 等人，2024 年)	众包	数据集	封闭的	微任务，复杂的多模态布局
VisualWebBench(Liu 等人，2024b)	网络	数据集	封闭的	光学字符识别 (OCR)，元素定位，动作预测
ScreenSpot(Cheng 等人，2024b)	截图	数据集	封闭的	直接从图像中进行点击/输入定位
WONDERBREAD(Wornow 等人，2024 年)	业务流程管理 (BPM) 任务	数据集	封闭的	工作流文档编制与改进
EnvDistraction(Ma 等人，2024 年)	合成图形用户界面 (GUI)	数据集	封闭的	衡量对干扰的鲁棒性
NaviQAte(Shahbandeh 等人，2024 年)	网络应用程序	数据集	封闭的	以问答形式进行的导航；功能引导
GAIA(Mialon 等人，2023 年)	通用的	数据集	开放的	开放词汇多模态问答
WebLINX(Lù 等人，2024 年)	实时网络	数据集	开放的	多轮对话导航
MiniWoB(Shi 等人，2017 年)	合成网络	环境	封闭的	低级鼠标/键盘技能
CompWoB(Furuta 等人，2023 年)	合成网络	环境	封闭的	组合式多步工作流
WebShop(Yao 等人，2022 年)	电子商务	环境	封闭的	遵循指令进行购物
WebArena(Zhou 等人，2023b)	自托管网络	环境	封闭的	长周期多领域任务
VisualWebArena(Koh 等人，2024a)	自托管网络	环境	封闭的	增加像素级多模态
WorkArena(Drouin 等人，2024 年)	网络，ServiceNow	环境	封闭的	企业知识工作用户界面
WorkArena++(Boisvert 等人，2024 年)	网络，ServiceNow	环境	封闭的	具有更难任务的 WorkArena
BrowserGym(Chezelles 等人，2024 年)	网络	环境	封闭的	由其他网络代理基准测试组成的统一健身房环境
ST - WebAgentBench(Levy 等人，2024 年)	自托管网络	环境	封闭的	安全可信度指标
VideoWebArena(Jang 等人，2024 年)	视频 + 网络	环境	封闭的	长上下文多模态推理
操作系统世界 (OSWorld)(谢等人，2024 年)	Windows 图形用户界面 (GUI)	环境	封闭的	桌面操作系统交互
Windows 代理竞技场 (WindowsAgentArena)(博纳蒂等人，2024 年)	Windows 图形用户界面 (GUI)	环境	封闭的	跨应用 Windows 任务基准测试
网络视觉语言导航 (WebVLN)(陈等人，2024 年)	实时网络	环境	开放的	视觉语言导航
网络旅行者 (WebVoyager)(何等人，2024b)	15 个实时网站	环境	开放的	端到端导航；HTML + 截图
自动网络基准测试 (AutoWebBench)(赖等人，2024 年)	实时网络	环境	开放的	强化学习微调，HTML 简化
多模态交互 (MMInA)(张等人，2024e)	实时网络	环境	开放的	多跳、多模态目标
网络画布 (WebCanvas)(潘等人，2024 年)	实时网络	环境	开放的	动态评估；界面变化适应性

Table 1: Benchmarks for GUI-agent research discussed in Section 3. ”Type” distinguishes static datasets from interactive environments; ”World” marks closed- vs. open-world assumptions.

表 1: 第 3 节中讨论的 GUI 代理研究基准。“类型”区分静态数据集与交互环境；“世界”标示封闭世界与开放世界假设。

Perception Modality	Data Type	Key Advantages	Key Limitations
Accessibility-Based 2) Resilient to minor layout changes 3) Lower privacy risk 2) May not handle highly dynamic or custom-drawn elements	Structured hierarchy (accessibility APIs) 1) Requires correct developer implementation	1) Offers semantic roles/labels	
HTML/DOM-Based 2) Directly targets interface elements 2) Needs careful preprocessing (e.g., snippet extraction, heuristics)	Hierarchical data (DOM tree) 1) HTML can be noisy/redundant	1) Rich structural information for web-based UIs	
Screen-Visual-Based 2) Handles custom visuals or games 2) Potential privacy concerns (full screenshot capture)	Pixel data (screenshots) 1) Higher computational overhead	1) Universal approach (no reliance on APIs)	
Hybrid (Multiple Modalities) 2) Better coverage in complex or dynamic tasks 2) Requires synchronizing data from multiple modalities	Combination (e.g., accessibility + DOM + Screen) 1) Increased system complexity	1) More robust to missing/incomplete data	

感知模态	数据类型	主要优势	主要局限
基于无障碍访问的 2) 对微小布局变化具有弹性 3) 隐私风险较低 2) 可能无法处理高度动态或自定义绘制的元素	结构化层次结构 (无障碍访问应用程序编程接口) 1) 需要开发者正确实现	1) 提供语义角色/标签	
基于超文本标记语言/文档对象模型的 2) 直接针对界面元素 2) 需要仔细的预处理 (例如，代码片段提取、启发式方法)	分层数据 (文档对象模型树) 1) 超文本标记语言可能存在噪声/冗余	1) 为基于网络的用户界面提供丰富的结构信息	
基于屏幕视觉的 2) 可处理自定义视觉效果或游戏 2) 存在潜在的隐私问题 (完整屏幕截图捕获)	像素数据 (屏幕截图) 1) 计算开销较高	1) 通用方法 (不依赖应用程序编程接口)	
混合 (多种模态) 2) 在复杂或动态任务中覆盖范围更广 2) 需要同步来自多种模态的数据	组合 (例如，无障碍访问 + 文档对象模型 + 屏幕) 1) 系统复杂度增加	1) 对缺失/不完整数据更具鲁棒性	

Table 2: Overview of Perception Modalities

表 2: 感知方式概述

Modality	Typical Scenarios	Example References
Accessibility-Based - Automated UI testing/checks	- Desktop/mobile apps with accessibility layers OS-based accessibility APIs, Official guidelines	
HTML/DOM-Based - Web scraping/search	- Web automation tasks (form-filling, data entry) Mind2Web, WebAgent, AutoWebGLM	
Screen-Visual-Based - Environments with no structured metadata	- Image-centric or game UIs OmniParser	
Hybrid - High-value scenarios (e.g., financial dashboards)	- Complex multi-step tasks OS-Atlas, UGround	

模态	典型场景	示例参考
基于无障碍 - 自动化 UI 测试/检查	- 带无障碍层的桌面/移动应用 基于操作系统的无障碍 API，官方指南	
基于 HTML/DOM - 网络爬取/搜索	- 网络自动化任务 (表单填写、数据录入) Mind2Web，WebAgent，AutoWebGLM	
基于屏幕视觉 - 无结构化元数据的环境	- 以图像为中心或游戏用户界面 OmniParser	
混合型 - 高价值场景 (例如，金融仪表盘)	- 复杂的多步骤任务 OS-Atlas，UGround	

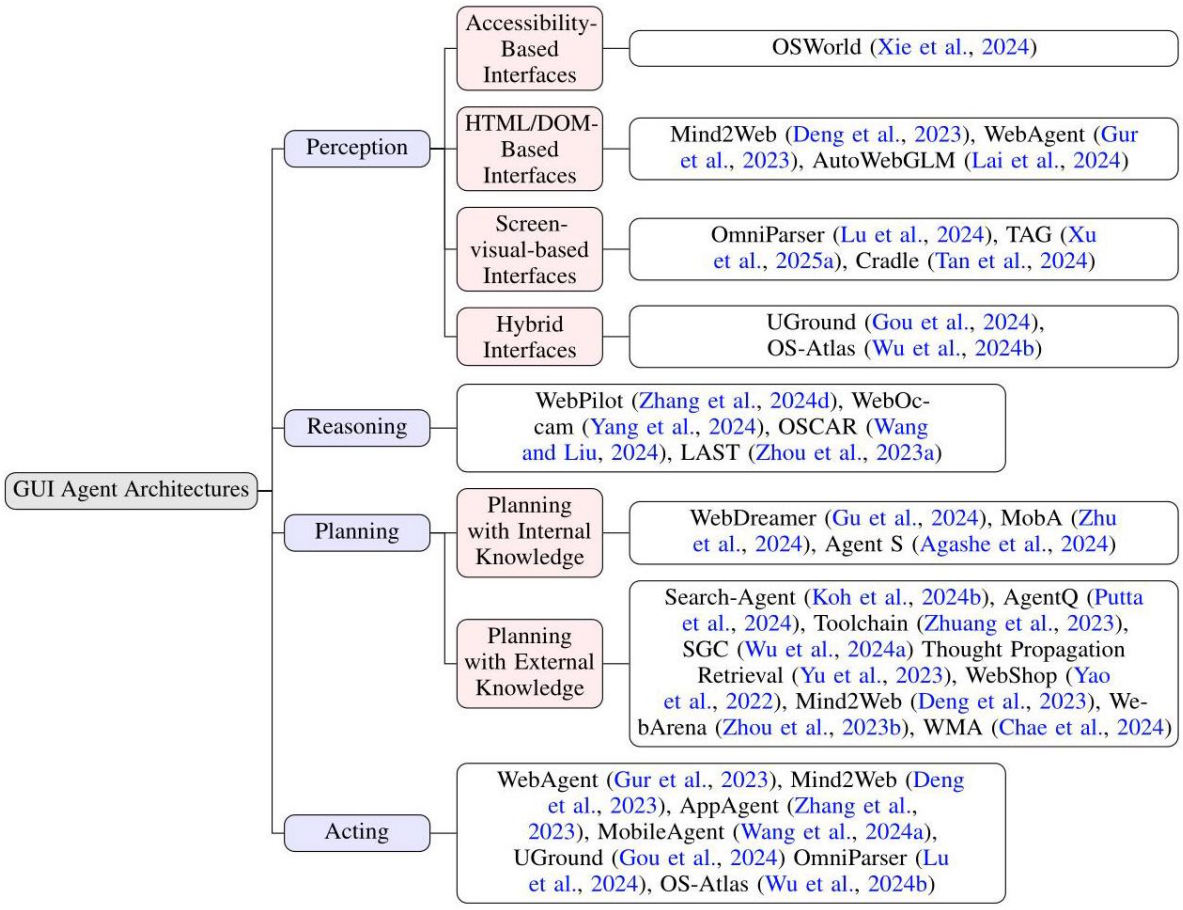

Figure 1: Taxonomy of GUI agent architectures.

图 1: 图形用户界面代理架构分类。

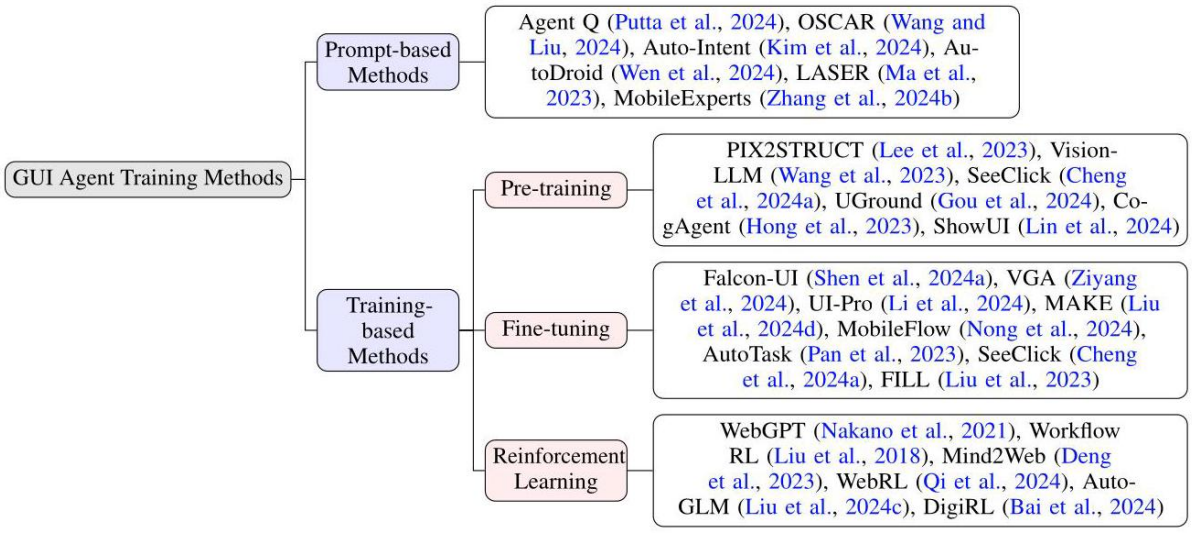

Figure 2: Taxonomy of GUI agent training methods.

图 2: 图形用户界面代理训练方法分类。