Mobile-Agent-E: Self-Evolving Mobile Assistant for Complex Tasks

Mobile-Agent-E: 面向复杂任务的自我进化移动助手

Zhenhailong Wang *[1] Haiyang Xu * Junyang Wang [2] Xi Zhang [2]

王振海龙 *[1] 徐海洋 * 王俊阳 [2] 张曦 [2]

Ming Yan [2] Ji Zhang [2] Fei Huang [2] Heng Ji * 1

闫明 [2] 张姬 [2] 黄飞 [2] 姬恒 * 1

Figure 1. We propose Mobile-Agent-E, a novel hierarchical multi-agent mobile assistant that outperforms previous state-of-the-art approaches (Zhang et al., 2023; Wang et al., 2024b;a) on complex real-world tasks. Mobile-Agent-E disentangles high-level planning and low-level action decision with dedicated agents. Equipped with a newly introduced self-evolution module that learns general Tips and reusable Shortcuts from past experiences, Mobile-Agent-E demonstrates further improvements in both performance and efficiency.

图 1. 我们提出了 Mobile-Agent-E，一种新颖的分层多智能体移动助手，在复杂的现实任务中优于以往最先进的方法 (Zhang et al., 2023；Wang et al., 2024b;a)。Mobile-Agent-E 通过专门的智能体将高层规划与低层动作决策解耦。配备了新引入的自我进化模块，该模块从过去经验中学习通用技巧 (Tips) 和可复用快捷操作 (Shortcuts)，Mobile-Agent-E 在性能和效率上均表现出进一步提升。

Abstract

摘要

Smartphones have become indispensable in modern life, yet navigating complex, multi-step tasks on mobile devices often remains frustrating and time-consuming. Recent advancements in large multimodal model (LMM)-based mobile agents have demonstrated the ability to perceive and act in mobile environments on behalf of users. However, current approaches face significant limitations: they fall short in addressing real-world human needs, struggle with reasoning-intensive and long-horizon tasks, and lack mechanisms to learn and improve from prior experiences. To overcome

智能手机已成为现代生活中不可或缺的工具，但在移动设备上处理复杂的多步骤任务仍常令人感到沮丧且耗时。基于大型多模态模型 (LMM) 的移动智能体的最新进展已展示出能够代表用户感知和操作移动环境的能力。然而，现有方法存在显著局限: 它们难以满足现实人类需求，难以应对推理密集型和长远任务，且缺乏从先前经验中学习和改进的机制。为克服

Preprint.

预印本。

these challenges, we introduce Mobile-Agent-E, a hierarchical multi-agent framework capable of self-evolution through past experience. By ”hierarchical,” we mean an explicit separation of high-level planning and low-level action execution. The framework comprises a Manager, responsible for devising overall plans by breaking down complex tasks into subgoals, and four subordinate agents—Perceptor, Operator, Action Reflector, and Notetaker-which handle fine-grained visual perception, immediate action execution, error verification, and information aggregation, respectively. Mobile-Agent-E also features a novel self-evolution module which maintains a persistent long-term memory comprising Tips and Shortcuts. Tips are general guidance and lessons learned from prior tasks on how to effectively interact with the environment. Shortcuts are reusable, executable sequences of atomic operations tailored

这些挑战，我们引入了 Mobile-Agent-E，一种能够通过过去经验自我进化的分层多智能体框架。所谓“分层”，指的是明确区分高层规划与低层动作执行。该框架包括一个管理者 (Manager)，负责通过将复杂任务分解为子目标来制定整体计划，以及四个下属智能体——感知者 (Perceptor)、操作员 (Operator)、动作反思者 (Action Reflector) 和记录者 (Notetaker)——分别处理细粒度视觉感知、即时动作执行、错误验证和信息汇总。Mobile-Agent-E 还具备一个新颖的自我进化模块，维护包含技巧 (Tips) 和快捷操作 (Shortcuts) 的持久长期记忆。技巧是从先前任务中学到的关于如何有效与环境交互的通用指导和经验教训。快捷操作是针对特定子程序的可复用、可执行的原子操作序列。

---

[1] University of Illinois Urbana-Champaign [2] Alibaba Group. *Corresponding authors: Zhenhailong Wang <wangz3@illinois.edu>, Haiyang Xu <shuofeng.xhy@alibaba-inc.com>, Heng Ji <hengji@illinois.edu>.

[1] 伊利诺伊大学厄巴纳-香槟分校 [2] 阿里巴巴集团。* 通讯作者: 王振海龙 <wangz3@illinois.edu>，徐海洋 <shuofeng.xhy@alibaba-inc.com>，姬恒 <hengji@illinois.edu>。

---

for specific subroutines. The inclusion of Tips and Shortcuts facilitates continuous refinement of task performance and efficiency. Alongside this framework, we introduce Mobile-Eval-E, a new benchmark featuring complex real-world mobile tasks requiring long-horizon, multi-app interactions. Empirical results show that Mobile-Agent-E achieves a 22% absolute improvement over previous state-of-the-art approaches across three foundation

model backbones. Additionally, we provide a comprehensive analysis of the impact of our self-evolution mechanism and suggest directions for future work. Code and data are publicly available for research purposes at https://x-plug.github.io/MobileAgent.

技巧和快捷操作的引入促进了任务性能和效率的持续优化。与此同时，我们推出了 Mobile-Eval-E，这是一个包含需要长远规划和多应用交互的复杂现实移动任务的新基准。实证结果显示，Mobile-Agent-E 在三种基础模型骨干上相较于以往最先进方法实现了 22% 的绝对提升。此外，我们还对自我进化机制的影响进行了全面分析，并提出了未来工作的方向。代码和数据已公开，供研究使用，网址:https://x-plug.github.io/MobileAgent。

1. Introduction

1. 引言

Smartphones have become integral to our daily lives, transforming the way we connect, work, and find entertainment. Yet, the average 4.5 hours people spend on their phones daily* often includes moments of frustration. Tedious tasks, such as deal hunting across multiple apps or gathering scattered information from various websites, often make us wish for a smart mobile assistant to ease these burdens. Recent advancements in large multimodal models (LMMs) (OpenAI, 2024; Anthropic, 2024; Team et al., 2024) have led to the emergence of LMM-based GUI agents (Wang et al., 2024c; Nguyen et al., 2024) capable of perceiving and acting in the Web, PC, and mobile environments on behalf of human users. Despite these initial successes, current research on mobile agents (Wang et al., 2024b; Zhang et al., 2023; Wang et al., 2024a; Li et al., 2024) has yet to fully address the challenges of real-world mobile tasks. We identify two key limitations below.

智能手机已成为我们日常生活的核心，改变了我们的连接、工作和娱乐方式。然而，人们平均每天花费在手机上的 4.5 小时 * 中，常伴随着挫败感。诸如在多个应用间寻找优惠或从各类网站收集分散信息等繁琐任务，常让我们渴望有一个智能移动助手来减轻负担。大型多模态模型 (LMM)(OpenAI, 2024；Anthropic, 2024；Team et al., 2024) 的最新进展催生了基于 LMM 的图形用户界面 (GUI) 智能体 (Wang et al., 2024c；Nguyen et al., 2024)，它们能够代表用户在网页、PC 和移动环境中感知和操作。尽管取得了初步成功，当前关于移动智能体的研究 (Wang et al., 2024b；Zhang et al., 2023；Wang et al., 2024a；Li et al., 2024) 尚未完全解决现实移动任务的挑战。我们归纳了以下两个主要限制。

First, we observe a significant gap between the capabilities of current mobile agents and the demands of real-world scenarios. While existing mobile agent tasks are typically short, straightforward, and goal-oriented, such as ”Navigate to a nearby gas station” (Wang et al., 2024a), tasks that better reflect actual human needs are far more complex. These tasks often require a combination of (1) intensive reasoning to address multiple constraints, such as balancing various factors or criteria; (2) long-horizon planning, which may involve a lengthy sequence of steps across multiple apps; and (3) exploration, where the instructions can be vague and require active information gathering rather than following a fixed trajectory. For instance, as shown in Figure 1, online shopping often involves navigating across different apps to compare prices and find the best deal. Furthermore, the highly dynamic nature of mobile environments, charac-

首先，我们观察到当前移动代理的能力与现实场景需求之间存在显著差距。现有的移动代理任务通常较短、简单且目标明确，例如“导航到附近的加油站”(Wang et al., 2024a)，而更能反映实际人类需求的任务则复杂得多。这些任务通常需要结合 (1) 密集推理以应对多重约束，如平衡各种因素或标准；(2) 长远规划，可能涉及跨多个应用的长序列步骤；(3) 探索，指令可能模糊，需要主动收集信息而非遵循固定路径。例如，如图 1 所示，网购常常涉及跨不同应用比较价格以寻找最佳优惠。此外，移动环境高度动态的特性，表现为弹出广告和频繁变化的应用布局，给解决这些复杂现实任务带来了额外挑战。

terized by pop-up advertisements and frequently changing app layouts, poses additional challenges in tackling these complex real-world tasks.

移动环境高度动态的特性，表现为弹出广告和频繁变化的应用布局，给解决这些复杂现实任务带来了额外挑战。

Second, unlike humans, who quickly adapt and become proficient with new devices or apps, current mobile agents lack the ability to learn from prior experiences. For example, when a human user first opens an app like Maps, it may take some trial and error to understand the layout and successfully perform a search. However, with each interaction, the user learns, becoming faster and more accurate the next time. In contrast, existing mobile agents treat every task as if it were their first attempt, allocating the same computational resources at each step and repeating the same mistakes, regardless of how many times they perform the same task. This inability to accumulate knowledge and refine actions from past experiences severely limits their ability to handle the aforementioned complex, long-horizon tasks, where subroutines such as searching and creating notes are often shared across different objectives.

其次，与人类能够快速适应并熟练使用新设备或应用不同，当前移动代理缺乏从以往经验中学习的能力。例如，当人类用户首次打开地图 (Maps) 应用时，可能需要通过反复试验来了解布局并成功完成搜索，但每次交互后，用户都会学习，下一次操作变得更快更准确。相比之下，现有移动代理将每个任务视为首次尝试，在每一步分配相同的计算资源并重复相同错误，无论执行多少次相同任务。这种无法积累知识和从过去经验中优化行为的缺陷，严重限制了它们处理上述复杂长远任务的能力，而这些任务中的子程序如搜索和创建笔记常常在不同目标间共享。

To address these limitations, we propose Mobile-Agent-E, a hierarchical multi-agent framework capable of self-evolution through past experiences. Mobile-Agent-E explicitly disentangles high-level planning-such as decomposing a task into smaller subgoals-from low-level actions, which involves determining specific actions and their parameters (e.g., tap (x, y)). The framework is structured with a Manager, responsible for creating overall plans, and four subordinate agents—Perceptor, Operator, Action Reflector, and Notetaker—that handle fine-grained visual perception, action decision, outcome verification, and information aggregation, respectively. This hierarchical design significantly enhances long-term planning and improves error recovery in complex tasks. Figure 1 shows an overview of Mobile-Agent-E on a challenging online shopping task requiring multi-step reasoning and interaction across three different apps.

为了解决这些限制，我们提出了 Mobile-Agent-E，一种能够通过过去经验自我进化的分层多代理框架。Mobile-Agent-E 明确区分了高层规划——如将任务分解为更小的子目标——与低层动作决策，即确定具体动作及其参数 (例如，点击 (x, y))。该框架由一个负责制定整体计划的管理者 (Manager) 和四个下属代理组成——感知者 (Perceptor)、操作员 (Operator)、动作反思者 (Action Reflector) 和笔记者 (Notetaker)，分别负责细粒度视觉感知、动作决策、结果验证和信息汇总。此分层设计显著增强了长期规划能力并改善了复杂任务中的错误恢复。图 1 展示了 Mobile-Agent-E 在一项需要多步推理和跨三个不同应用交互的复杂网购任务中的概览。

Mobile-Agent-E also features a self-evolution module, which includes a persistent long-term memory and two Experience Reflectors. We define two types of critical knowledge that are continuously updated in the long-term memory across tasks: Tips-general guidance on effective interactions and lessons learned from previous trail-and-error experiences—and Shortcuts—reusable, executable functions that contains sequences of atomic operations tailored to efficiently complete recurring subroutines under specific preconditions. After completing each task, the Experience Reflectors are triggered to update the Tips and propose new Shortcuts based on the interaction history. These are then fed to the Manager and Operator, enabling improved planning and action decision-making in future tasks. This design draws inspiration from human cognitive science, where Tips are akin to the lessons encoded in episodic memory (Tul-

Mobile-Agent-E 还具备自我进化模块，包括持久的长期记忆和两个经验反思器。我们定义了两类关键知识，持续在长期记忆中跨任务更新: 提示 (Tips)——关于有效交互的一般指导和从以往反复试验中获得的经验教训；快捷方式 (Shortcuts)——可重用的可执行函数，包含针对特定前提条件下高效完成重复子程序的原子操作序列。每完成一项任务，经验反思器便被触发，更新提示并基于交互历史提出新的快捷方式。这些内容随后反馈给管理者和操作员，提升未来任务中的规划和动作决策能力。该设计灵感来源于人类认知科学，其中提示类似于编码在情景记忆 (episodic memory)(Tulving, 2002) 中的经验教训，涉及回忆特定过去经历并用于指导未来决策，而快捷方式则类似于程序性知识，有助于高效且常常是无意识地执行熟练任务 (Squire & Zola, 1996；Anderson, 1982)。图 1 中提供了快捷方式和提示的示例。

---

*https://explodingtopics.com/blog/ smartphone-usage-stats

*https://explodingtopics.com/blog/ smartphone-usage-stats

---

ving, 2002), which involves recalling specific past experiences and using them to inform future decisions, while Shortcuts resemble procedural knowledge that facilitates the efficient and often subconscious execution of well-practiced tasks (Squire & Zola, 1996; Anderson, 1982). An example of Shortcuts and Tips is provided in Figure 1.

ving, 2002)，这涉及回忆特定过去经历并用于指导未来决策，而快捷方式则类似于程序性知识，有助于高效且常常是无意识地执行熟练任务 (Squire & Zola, 1996；Anderson, 1982)。图 1 中提供了快捷方式和提示的示例。

To address the limitation of existing mobile benchmarks, which mainly include short-horizon and straightforward tasks with already saturated performance, we introduce a new benchmark, Mobile-Eval-E, designed to

evaluate complex, real-world tasks. Mobile-Eval-E features more than twice the number of expected operations per task compared to previous benchmarks (Wang et al., 2024b; Zhang et al., 2023; Wang et al., 2024a) and incorporating a significantly higher proportion of tasks requiring multi-app interactions. Accompanying the benchmark, we introduce a new evaluation metric called the Satisfaction Score to address the challenge posed by real-world tasks that often lack a binary success flag or a ground truth trajectory. This metric is computed based on human-written rubrics that account for both milestone completion, such as ”opened Maps,” and exploratory behaviors, such as ”viewed more than one review.” This approach offers a reliable measure of agent performance aligned with human preferences. We further propose a Satisfaction Score vs Steps (SSS) curve to better evaluate and visualize the efficiency of mobile agents. Mobile-Eval-E sets a high standard of difficulty, with prior state-of-the-art methods achieving only about 50-70% of human satisfaction.

为解决现有移动基准测试主要包含短期且简单任务且性能已趋于饱和的局限，我们引入了新基准 Mobile-Eval-E,用于评估复杂的现实任务。Mobile-Eval-E 的每项任务预期操作次数是以往基准 (Wang et al., 2024b；Zhang et al., 2023；Wang et al., 2024a) 的两倍以上，且包含显著更高比例的多应用交互任务。配套基准，我们提出了一种新的评估指标——满意度得分 (Satisfaction Score)，以应对现实任务常缺乏二元成功标志或标准轨迹的挑战。该指标基于人工编写的评分标准，兼顾里程碑完成情况 (如“打开地图”) 和探索行为 (如“查看多条评论”)，提供与人类偏好一致的可靠性能衡量。我们进一步提出满意度得分与步骤数 (SSS) 曲线，以更好地评估和可视化移动代理的效率。Mobile-Eval-E 设定了较高难度标准，现有最先进方法仅能达到约 50-70% 的人工满意度。

Empirical results show that Mobile-Agent-E achieves an average absolute gain of 22.1% over previous state-of-the-art approaches across three different foundation model backbones. Mobile-Agent-E also demonstrates promising self-evolution behavior in both performance and efficiency, resulting in a 6.5% absolute improvement compared to no evolution. The incorporation of Shortcuts further reduces the computational overhead, achieving speeds comparable to prior models while delivering significantly better performance. Additionally, we provide a comprehensive analysis of various aspects of self-evolution’s impact and outline directions for future work.

实证结果表明，Mobile-Agent-E 在三种不同的基础模型骨干网络上，相较于之前的最先进方法，平均绝对提升了 22.1%。Mobile-Agent-E 还展示了在性能和效率方面的有希望的自我进化行为，相较于无进化方案实现了 6.5% 的绝对提升。引入 Shortcuts 进一步降低了计算开销，实现了与先前模型相当的速度，同时显著提升了性能。此外，我们还对自我进化影响的各个方面进行了全面分析，并概述了未来工作的方向。

2. Mobile-Agent-E

2. Mobile-Agent-E

Figure 2 provides an overview of Mobile-Agent-E. A summary of the notation definitions is presented in Table 1. We detail the hierarchical multi-agent framework (§2.1) and the self-evolution module (§2.2) in Mobile-Agent-E below.

图 2 展示了 Mobile-Agent-E 的整体架构。符号定义汇总见表 1。以下我们详细介绍 Mobile-Agent-E 中的分层多智能体框架 (§2.1) 和自我进化模块 (§2.2)。

Table 1. Notation definitions.

表 1. 符号定义。

	Notation Description
Environment	
I	Input task query
a^t	Action † at time t
s^t	Phone state (screenshot) at time t
Agents	
$\mathcal{A}_P$	Perceptor
$\mathcal{A}_M$	Manager
$\mathcal{A}_O$	Operator
$\mathcal{A}_R$	Action Reflector
$\mathcal{A}_N$	Notetaker
$\mathcal{A}_{ES}$	Experience Reflector for Shortcuts
$\mathcal{A}_{ET}$	Experience Reflector for Tips
Working Memory	
W_V^t	Visual perception result at time t
W_P^t	Overall plan (decomposed subgoals) at time t
W_S^t	Current subgoal at time t
W_G^t	Progress status at time t
W_N^t	Important notes at time t
W_{EF}^t	Error Escalation Flag at time t
$\mathbf{W_A}$	Action history with outcome status
$\mathbf{W_E}$	Error history with feedback
Long-term Memory	
L_S	Shortcuts
L_T	Tips

	符号说明
环境	
I	输入任务查询
a^t	时间 t 的操作 †
s^t	时间 t 的手机状态 (截图)
代理	
$\mathcal{A}_P$	感知者
$\mathcal{A}_M$	管理者
$\mathcal{A}_O$	操作员
$\mathcal{A}_R$	动作反射器
$\mathcal{A}_N$	记录员
$\mathcal{A}_{ES}$	快捷方式经验反思器
$\mathcal{A}_{ET}$	提示经验反思器
工作记忆	
W_V^t	时间 t 的视觉感知结果
W_P^t	时间 t 的整体计划 (分解子目标)
W_S^t	时间 t 的当前子目标
W_G^t	时间 t 的进度状态
W_N^t	时间 t 的重要备注
W_{EF}^t	时间 t 的错误升级标志
$\mathbf{W_A}$	带结果状态的操作历史
$\mathbf{W_E}$	带反馈的错误历史
长期记忆	
L_S	快捷方式
L_T	提示

2.1. Hierachical Multi-Agent Framework

2.1. 分层多智能体框架

Figure 3 provides a detailed breakdown of the main agent loop with concrete examples. Except for the Perceptor, all reasoning agents are instantiated from a frozen large multimodal model (LMM), such as GPT-40 (OpenAI, 2024). The inputs and outputs of each agent are detailed as follows.

图 3 详细展示了主智能体循环的具体示例。除感知器 (Perceptor) 外，所有推理智能体均由冻结的大型多模态模型 (LMM)，如 GPT-40(OpenAI，2024) 实例化。每个智能体的输入和输出详述如下。

Manager $(\mathcal{A}_M)$: High-level planning. The Manager focuses on devising high-level plans to achieve the user's requests. At each step, the Manager checks the input query I , the current screenshot s_t , the previous overall plan W_P^{t-1} , the previous subgoal W_S^{t-1} , the progress status W_G^{t-1} , available Shortcuts from long-term memory L_S , and any recorded important notes W_N^{t-1} to provide an updated overall plan W_P^t and identify the next immediate subgoal W_S^t to achieve. Note that the Manager does not condition on the fine-grained perception results from the Perceptor, as it is not necessary and can add noise to high-level planning.

管理者 $(\mathcal{A}_M)$: 高层规划。管理者专注于制定实现用户请求的高层计划。在每一步，管理者会检查输入查询 I 、当前截图 s_t 、之前的整体计划 W_P^{t-1} 、之前的子目标 W_S^{t-1} 、进度状态 W_G^{t-1} 、来自长期记忆的可用快捷方式 L_S 以及任何记录的重要笔记 W_N^{t-1} ，以提供更新后的整体计划 W_P^t 并确定下一个立即要实现的子目标 W_S^t 。注意，管理者不会基于感知器的细粒度感知结果进行条件判断，因为这既非必要，也可能为高层规划带来噪声。

(1)

$$W_P^t, W_S^t = \mathcal{A}_M\left(I, s_t, W_P^{t-1}, W_S^{t-1}, W_G^{t-1}, W_N^{t-1}, L_S,\right. \\ \left.\mathbf{W_E}\left[-k:\right]\right) \quad \text{if } t \geq k \text{ and } W_{EF}^{t-1} == \text{ True} \tag{2}$$

---

†a_t can represent either a single atomic operation or a sequence of atomic operations if performing a Shortcut.

†a_t 可以表示单个原子操作，也可以表示执行快捷方式时的一系列原子操作。

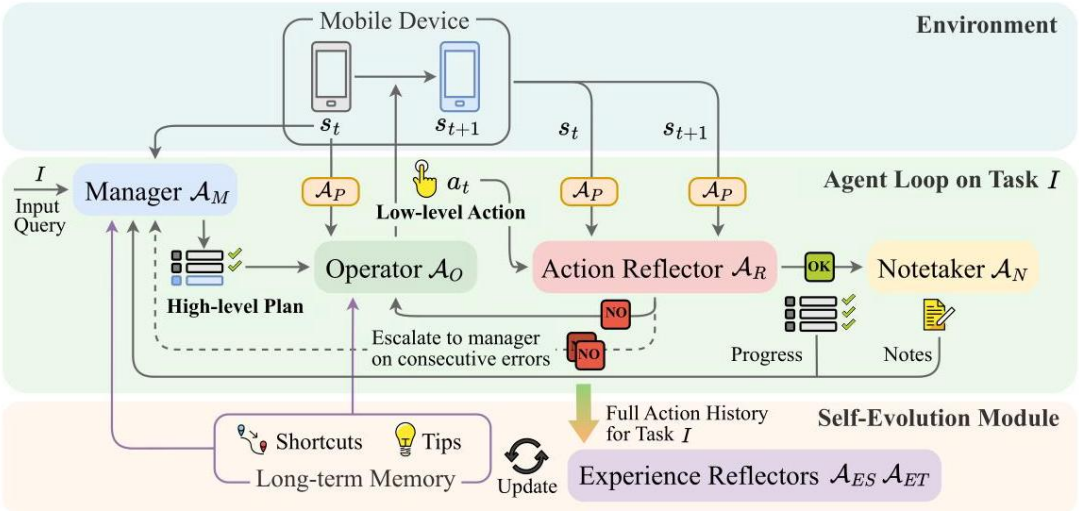


Figure 2. An overview of the Mobile-Agent-E framework, where the Manager, Perceptor $(\mathcal{A}_P)$, Operator, Action Reflector, and Notetaker are involved in the main agent loop for each task, while two Experience Reflectors contribute to updating long-term memory across tasks. Decision-making at each step is disentangled into high-level planning by the Manager and low-level actions by the Operator. The Action Reflector verifies the outcome of each action, tracks progress, and provides error feedback. The Notetaker aggregates important information during navigation. A detailed example illustrating one step in the agent loop and the self-evolution process is presented in Figures 3 and 4.

图 2. Mobile-Agent-E 框架概览，其中管理者、感知器 $(\mathcal{A}_P)$ 、操作员、动作反思器和笔记员参与每个任务的主智能体循环，而两个经验反思器则协助跨任务更新长期记忆。每一步的决策被拆分为管理者负责的高层规划和操作员负责的低层动作。动作反思器验证每个动作的结果，跟踪进度并提供错误反馈。笔记员在导航过程中汇总重要信息。图 3 和图 4 展示了智能体循环中一步操作及自我进化过程的详细示例。

Additionally, when the model is potentially stuck in an error loop, that is, observing k consecutive failed actions (e.g., $k = 2$) reported by the Action Reflector, a special Error Escalation Flag W_{EF}^{t-1} will be raised to the Manager. In such cases, the Manager will be prompted with additional information about the recent errors $\mathbf{W_E}\left[-k:\right]$ and asked to determine how to address the error from a higher-level perspective-such as refining the overall plan or adjusting the current subgoal to rectify the issue. In other cases, when an error first occurs, the Operator will attempt to address it before escalating the issue to the Manager. A concrete example of how the error escalation can help recovering from errors can be found in Figure 9.

此外，当模型可能陷入错误循环时，即观察到动作反思器报告的 k 次连续失败动作 (例如，$k = 2$)，将向管理者发出特殊的错误升级标志 W_{EF}^{t-1} 。在这种情况下，管理者会收到关于近期错误的额外信息 $\mathbf{W_E}\left[-k:\right]$ ，并被要求从更高层次视角决定如何处理错误——例如，优化整体计划或调整当前子目标以纠正问题。在其他情况下，当错误首次发生时，操作员会尝试解决，若未果则将问题升级至管理者。图 9 中有一个具体示例展示了错误升级如何帮助恢复错误。

Perceptor $(\mathcal{A}_P)$: Fine-grained visual perception. The Perceptor aims to detect and recognize rich information about the current phone state, such as icons and text. We use a purely vision-based perception module that does not rely on the underlying XML file, following (Wang et al., 2024a). The Perceptor consists of three main components: an OCR model, an icon grounding model, and an icon captioning model. Given a screenshot s_t at time t , the Perceptor generates a fine-grained list of texts and icons, along with their corresponding coordinates W_V^t . Note that we still provide the original screenshot image to subsequent reasoning agents as a holistic visual context.

感知器 $(\mathcal{A}_P)$: 细粒度视觉感知。感知器旨在检测和识别当前手机状态的丰富信息，如图标和文本。我们采用纯视觉感知模块，不依赖底层 XML 文件，参照 (Wang et al., 2024a)。感知器由三个主要组件组成:OCR 模型、图标定位模型和图标描述模型。给定时间点 t 的截图 s_t ，感知器生成细粒度的文本和图标列表及其对应坐标 W_V^t 。注意，我们仍将原始截图图像提供给后续推理智能体，作为整体视觉上下文。

$$W_V^t = \mathcal{A}_P\left(s_t\right) \tag{3}$$

Operator $(\mathcal{A}_O)$: Low-level action decisions. The Operator decides which concrete action to perform based on the input query I , the overall plan W_P^t and current subgoal W_S^t from the Manager, the previous progress status W_G^{t-1} , the important notes W_N^{t-1} , along with a history of the latest m actions $\mathbf{W_A}\left[-m:\right]$ and errors $\mathbf{W_E}\left[-m:\right]$.[‡] The action history includes both the action and its outcome (success or failure). The Operator is explicitly prompted to rectify errors if it observes unresolved failures in the history. The Operator also considers the Tips as guidance from the long-term memory, which can be self-evolved from past experiences. To enable accurate generation of the action parameters, e.g., the (x, y) coordinates on the screen for tapping, we also provide the Operator with the fine-grained perception results W_V^t from the Perceptor along with the screenshot s_t .

操作器 $(\mathcal{A}_O)$: 低层次动作决策。操作器根据输入查询 I 、整体计划 W_P^t 和管理器提供的当前子目标 W_S^t ，以及之前的进展状态 W_G^{t-1} 、重要备注 W_N^{t-1} ，并结合最新动作历史 m $\mathbf{W_A}\left[-m:\right]$ 和错误记录 $\mathbf{W_E}\left[-m:\right]$.[‡] ，决定执行具体动作。动作历史包括动作本身及其结果 (成功或失败)。如果历史中存在未解决的失败，操作器会被明确提示进行纠正。操作器还将提示视为来自长期记忆的指导，这些提示可通过过去经验自我演化而来。为了准确生成动作参数，例如屏幕上的 (x, y) 坐标以进行点击，我们还为操作器提供了来自感知器 (Perceptor) 的细粒度感知结果 W_V^t 及截图 s_t 。

$$a_t = \mathcal{A}_O\left(I, s_t, W_V^t, W_P^t, W_S^t, W_G^t, W_N^t,\right.$$

$$\left.\mathbf{W_A}\left[-m:\right], \mathbf{W_E}\left[-m:\right], L_S, L_T\right) \tag{4}$$

The output of the Operator is the next action a_t to perform. The action space is defined to contain not only Atomic Operations but also Shortcuts, which can evolve through tasks. The atomic operations include Open_App, Tap, Swipe, Type, Enter, Switch_App, Back, Home, and Wait. The full descriptions of the atomic operations can be found in Table 8. We detail the definitions and examples of Shortcuts and Tips in §2.2 .

操作器的输出是下一步要执行的动作 a_t。动作空间不仅包含原子操作，还包括可通过任务演化的快捷方式。原子操作包括打开应用 (Open_App)、点击 (Tap)、滑动 (Swipe)、输入 (Type)、确认 (Enter)、切换应用 (Switch_App)、返回 (Back)、主页 (Home) 和等待 (Wait)。原子操作的完整描述见表 8。我们在 §2.2 中详细说明了快捷方式和提示的定义及示例。

---

*We empirically set $m = 5$ in our experiments.

* 我们在实验中经验性地设定了 $m = 5$。

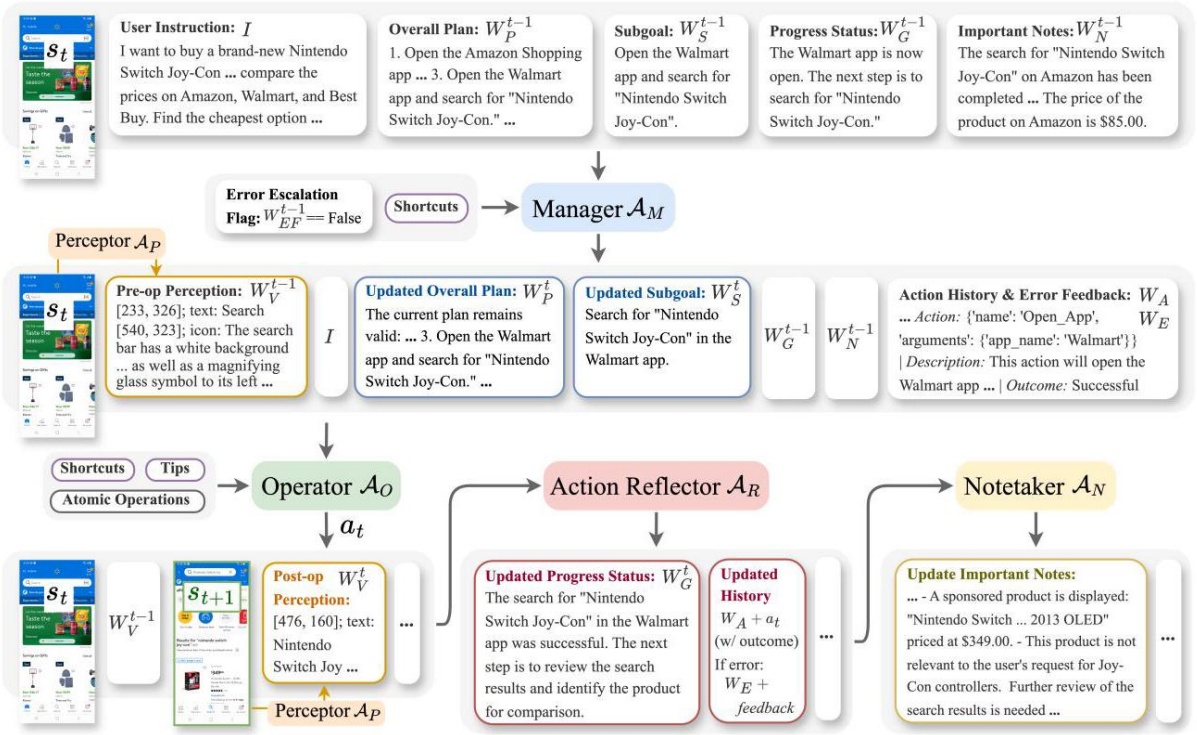


Figure 3. A detailed breakdown of one inference step t with Mobile-Agent-E, showing the inputs and outputs of each agent. Omitted information indicates no change.

图 3. 使用 Mobile-Agent-E 进行一次推理步骤 t 的详细分解，展示了各代理的输入和输出。省略的信息表示无变化。

Action Reflector ($\mathcal{A}_R$) : Reflection on the action outcome. The Action Reflector checks the screenshots before (s_t) and after (s_{t+1}) of an action (a_t) to verify if the previous action achieves the expected outcome. We define three types of outcomes for an action: A. Successful or partially successful: the result of the last action meets the expectation; § B. Failed: the last action results in a wrong page; and C. Failed: the last action produces no changes. After identifying the outcome, if the outcome is A, the Action Reflector updates the action history $\mathbf{W_A}[t]$ as well as the progress status W_G^t . If the outcome is B or C , the Action Reflector additionally provides a description of the error and suggests potential reasons and solutions in $\mathbf{W_E}[t]$.

动作反思器 ($\mathcal{A}_R$) : 对动作结果的反思。动作反思器检查动作前后的截图 (s_t) (s_{t+1}) ，以验证前一动作 (a_t) 是否达到了预期效果。我们定义了动作的三种结果类型:A. 成功或部分成功: 上一次动作的结果符合预期; § B. 失败: 上一次动作导致错误页面；C. 失败: 上一次动作未产生任何变化。识别结果后，若为 A，动作反思器会更新动作历史 $\mathbf{W_A}[t]$ 及进展状态 W_G^t 。若为 B 或 C ，动作反思器还会在 $\mathbf{W_E}[t]$ 中提供错误描述及可能的原因和解决方案建议。

$$W_V^{t+1} = \mathcal{A}_P\left(s_{t+1}\right) \quad \text{\#run Perceptor on } s_{t+1} \tag{5}$$

$$\mathbf{W_A}[t], \mathbf{W_E}[t], W_G^t = \mathcal{A}_R\left(I, s_t, W_V^t, s_{t+1}, W_V^{t+1},\right.$$

$$\left. a_t, W_S^t, W_G^{t-1},\right) \tag{6}$$

Notetaker ($\mathcal{A}_N$) : Information aggregation. In complex mobile tasks, we often need to keep track of important notes during exploration, such as the price of a product or

笔记员 ($\mathcal{A}_N$) : 信息汇总。在复杂的移动任务中，我们常需在探索过程中记录重要信息，如商品价格或

the phone number of a restaurant. The Notetaker is dedicated to extracting and aggregating task-relevant information W_N^t after each step, based on the input query I , overall plan W_P^t , current subgoal W_S^t , current progress W_G^t , fine-grained screen perception W_V^{t+1} after executing the action, and existing notes W_N^{t-1} .

餐厅电话号码。笔记员专注于在每一步后提取并汇总与任务相关的信息 W_N^t ，基于输入查询 I 、整体计划 W_P^t 、当前子目标 W_S^t 、当前进展 W_G^t 、执行动作后的细粒度屏幕感知 W_V^{t+1} 及已有笔记 W_N^{t-1} 。

$$W_N^t = \mathcal{A}_N\left(I, s_{t+1}, W_V^{t+1}, W_P^t, W_S^t, W_G^t, W_N^{t-1}\right) \tag{7}$$

2.2. Self-Evolution Module

2.2. 自我演化模块

Inspired by how humans become increasingly effective and efficient in operating smartphones, we maintain a long-term memory that persists across tasks and leverage two dedicated agents to reflect on past experiences. The long-term memory contains two important types of knowledge to evolve upon, Tips and Shortcuts, aiming to

improve both the performance and efficiency of the agent. Figure 4 provides a detailed breakdown of one self-evolution step.

受人类在操作智能手机时变得越来越高效和有效的启发，我们保持一个跨任务持久的长期记忆，并利用两个专门的代理来反思过去的经验。长期记忆包含两种重要的知识类型，提示 (Tips) 和快捷方式 (Shortcuts)，旨在提升代理的性能和效率。图 4 详细展示了一个自我进化步骤的分解。

Tips (L_T) are defined as general guidance on effective interactions and lessons learned from previous trial-and-error experiences. Tips resemble episodic memory (Tulving, 2002), which enables humans to recall past experiences and apply insights to future decisions.

提示 (Tips) (L_T) 被定义为关于有效交互的一般指导和从以往反复试验中获得的经验教训。提示类似于情景记忆 (episodic memory)(Tulving, 2002)，使人类能够回忆过去的经历并将洞见应用于未来的决策。

---

§ Some actions may need multiple repetitions to fulfill the expectation, for example, swipe up to find reviews. Thus, we include partially successful as meeting the expectation.

§ 某些操作可能需要多次重复才能达到预期，例如向上滑动以查找评论。因此，我们将部分成功视为满足预期。

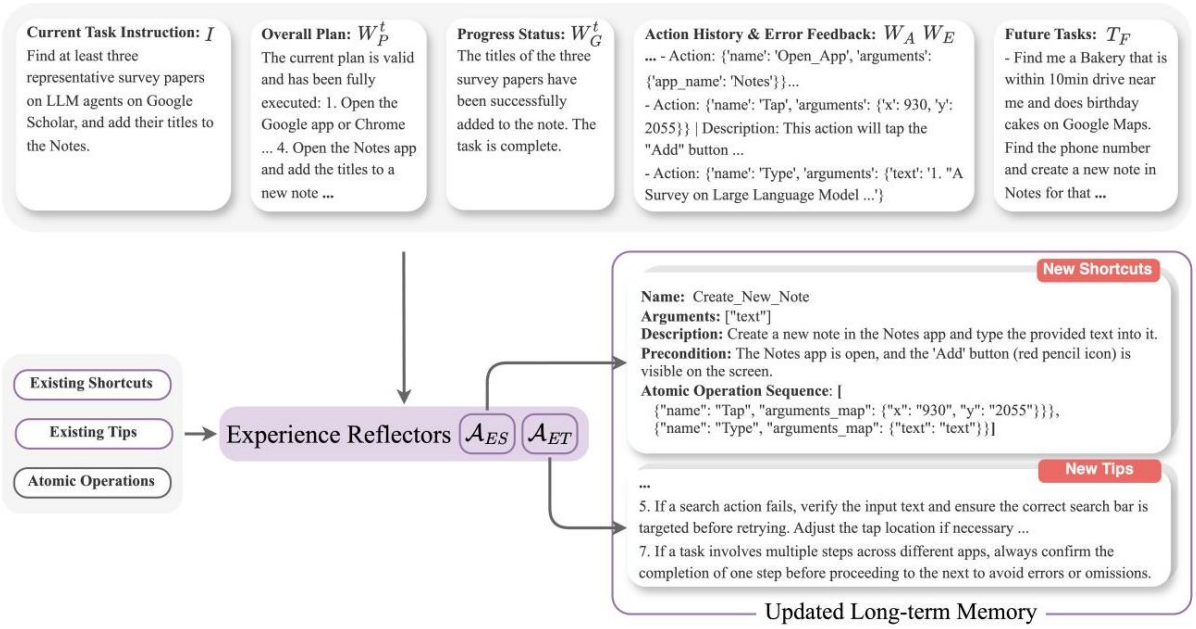


Figure 4. Illustration of the inputs and outputs to the Experience Reflectors for a single self-evolution step, including a concrete example of the newly generated Shortcuts and Tips.

图 4. 体验反思器 (Experience Reflectors) 在单次自我进化步骤中的输入和输出示意图，包括新生成的快捷方式和提示的具体示例。

Shortcuts (L_S) are defined as reusable, executable functions composed of sequences of atomic operations tailored for recurring subroutines. Shortcuts are akin to procedural knowledge, which allows humans to perform well-practiced tasks efficiently and often subconsciously (Squire & Zola, 1996; Anderson, 1982). Due to the highly dynamic nature of the mobile environment, a Shortcut may only be applicable in certain states. For instance, the "Tap-Type_and_Enter" Shortcut is usable only when the current screen has a text input box. To address this, we explicitly include a precondition in the definition of a Shortcut and require the Operator to verify that the current state satisfies the precondition before using the Shortcut. The arguments of a Shortcut have a unique one-to-one mapping to the arguments of its atomic operations.

快捷方式 (Shortcuts) (L_S) 被定义为由一系列原子操作组成的可重用、可执行函数，专为重复子程序量身定制。快捷方式类似于程序性知识 (procedural knowledge)，使人类能够高效且常常在无意识中完成熟练任务 (Squire & Zola, 1996；Anderson, 1982)。由于移动环境的高度动态性，快捷方式可能仅在特定状态下适用。例如，"Tap-Type_and_Enter"快捷方式仅在当前屏幕有文本输入框时可用。为此，我们在快捷方式定义中明确包含前置条件，并要求操作员在使用快捷方式前验证当前状态是否满足该前置条件。快捷方式的参数与其原子操作的参数一一对应。

When the self-evolution module is enabled, we leverage two Experience Reflectors, $\mathcal{A}_{ES}$ and $\mathcal{A}_{ET}$, to update the Tips and Shortcuts at the end of each task. The Experience Reflectors are also instantiated from frozen large multimodal model such as GPT-40. Let the final time step of a task be $t = \tau$. The input to the Experience Reflectors includes the input query I , the final overall plan W_P^τ , the final progress status W_G^τ , the entire action history $\mathbf{W_A}$ and error history $\mathbf{W_E}$, the existing Shortcuts L_S and Tips L_T , and a list of future tasks T_F (if provided). The outputs consist of newly generated Shortcuts in a predefined JSON format and updated Tips in natural language. Figures 12 and 13 shows a full list of generated Shortcuts and Tips by Mobile-Agent-E.

当自我进化模块启用时，我们利用两个体验反思器，$\mathcal{A}_{ES}$ 和 $\mathcal{A}_{ET}$ ，在每个任务结束时更新提示和快捷方式。体验反思器同样由冻结的大型多模态模型 (如 GPT-40) 实例化。设任务的最终时间步为 $t = \tau$ 。体验反思器的输入包括输入查询 I 、最终整体计划 W_P^τ 、最终进度状态 W_G^τ 、完整的动作历史 $\mathbf{W_A}$ 和错误历史 $\mathbf{W_E}$ 、现有快捷方式 L_S 和提示 L_T ，以及未来任务列表 T_F (如有提供)。输出包括以预定义 JSON 格式生成的新快捷方式和以自然语言更新的提示。图 12 和图 13 展示了 Mobile-Agent-E 生成的完整快捷方式和提示列表。

$$L_T = \mathcal{A}_{ET}\left(I, W_P^\tau, W_G^\tau, \mathbf{W_A}, \mathbf{W_E}, T_F, L_T\right) \tag{8}$$

$$L_S = \mathcal{A}_{ES}\left(I, W_P^\tau, W_G^\tau, \mathbf{W_A}, \mathbf{W_E}, T_F, L_S\right) \tag{9}$$

The updated Tips and Shortcuts are then utilized by the Manager and the Operator in the subsequent task, facilitating evolution in both high-level planning and low-level action decisions.

更新后的提示和快捷方式随后被管理器和操作员在后续任务中使用，促进高层规划和低层动作决策的双重进化。

3. Experiments

3. 实验

We perform a dynamic evaluation, evaluating the models in real-time and on actual devices—following previous work (Wang et al., 2024b; Zhang et al., 2023; Wang et al., 2024a). Specifically, we use the Android Debug Bridge (ADB) to control an Android phone ¶ and perform human evaluation on the recorded screenshots and action histories.

我们进行动态评估，实时在实际设备上评测模型——遵循先前工作 (Wang et al., 2024b；Zhang et al., 2023；Wang et al., 2024a)。具体而言，我们使用 Android 调试桥 (Android Debug Bridge，ADB) 控制一部 Android 手机 ¶，并对录制的截图和动作历史进行人工评估。

3.1.A More Challenging Benchmark: Mobile-Eval-E

3.1. 一个更具挑战性的基准:Mobile-Eval-E

Existing dynamic benchmarks (Wang et al., 2024b; Zhang et al., 2023; Wang et al., 2024a) primarily focus on short-horizon, straightforward tasks, where the performance has already saturated. To address this limitation, we propose a new and challenging benchmark, Mobile-Eval-E, which emphasizes reasoning-intensive, long-horizon, multi-app tasks. Mobile-Eval-E comprises 25 manually crafted tasks spanning 5 real-world scenarios: ”Restaurant Recom-

现有的动态基准 (Wang et al., 2024b；Zhang et al., 2023；Wang et al., 2024a) 主要关注短期、简单任务，性能已趋于饱和。为解决此限制，我们提出了一个新的挑战性基准 Mobile-Eval-E，强调推理密集型、长周期、多应用任务。Mobile-Eval-E 包含 25 个手工设计的任务，涵盖 5 个真实场景:“餐厅推荐-

---

[q] A Samsung Galaxy A15 is used for all experiments.

[q] 所有实验均使用三星 Galaxy A15 手机进行。

Table 2. Comparison with existing dynamic evaluation benchmarks on real devices. Mobile-Eval-E emphasizes long-horizon, complex tasks that require significantly more operations and a wider variety of apps.

表 2. 与现有动态评估基准在真实设备上的比较。Mobile-Eval-E 强调长周期、复杂任务，需显著更多操作和更广泛的应用种类。

Benchmark	#Tasks	#Multi-App Tasks	#Apps	Avg # Ops	Total # Ops
Mobile-Eval	33	3	10	5.55	183
Mobile-Eval-v2	44	4	10	5.57	245
AppAgent	45	0	9	6.31	284
Mobile-Eval-E	25	19	15	14.56	364

基准测试	任务数量	多应用任务	应用数量	平均操作数	总操作数
移动评估	33	3	10	5.55	183
移动评估-v2	44	4	10	5.57	245
应用代理	45	0	9	6.31	284
移动评估-E	25	19	15	14.56	364

mendation", "Information Searching", "Online Shopping", "What's Trending", and "Travel Planning". As shown in Table 2, Mobile-Eval-E significantly surpasses previous benchmarks in complexity, featuring more than 2× the number of expected operations per task and a greater total number of operations. Most tasks in existing benchmarks can be viewed as specific subgoals in Mobile-Eval-E. Additionally, Mobile-Eval-E encompasses a broader range of Apps, with 76% of the tasks requiring interactions with multiple Apps-compared to less than 10% in previous benchmarks. In §3, we demonstrate that this benchmark presents a substantial challenge for existing state-of-the-art models. The full set of task queries can be found in Appendix Table 7. Due to the long-horizon nature of the tasks, we keep the number of tasks relatively small to ensure a reasonable human evaluation workload for fine-grained analysis.

"推荐"、"信息搜索"、"在线购物"、"流行趋势"和"旅行规划"。如表 2 所示，Mobile-Eval-E 在复杂度上显著超越了以往的基准测试，任务预期操作次数超过 2× 倍，总操作次数也更多。现有基准测试中的大多数任务可视为 Mobile-Eval-E 中的具体子目标。此外，Mobile-Eval-E 涵盖了更广泛的应用程序，76% 的任务需要与多个应用程序交互，而以往基准测试中这一比例不足 10%。在第 3 节中，我们展示了该基准对现有最先进模型构成了重大挑战。完整的任务查询集见附录表 7。由于任务具有长远目标性质，我们保持任务数量相对较少，以确保细粒度分析的人类评估工作量合理。

3.2. Metrics with Better Human Alignment

3.2. 更符合人类评价的指标

Previous dynamic evaluation typically employs a binary success rate or a completion rate against a "ground truth" trajectory to evaluate the level of task completeness. However, real-world tasks often do not have a binary success flag or a single ground truth action sequence. For example, some tasks, such as "Plan a one-day itinerary for Palo Alto," may involve exploration and information aggregation, where multiple reasonable solutions might exist. Thus, we seek to measure human satisfaction rather than exact matches with a ground truth trajectory. For each task, we first manually write a list of rubrics (an example shown in Figure 5(a)), containing both milestone steps (e.g., "Opened Tripadvisor") and satisfaction criteria (e.g., "Viewed multiple attractions"). We then define the Satisfaction Score (SS) as the number of fulfilled rubrics divided by the total number of rubrics, as judged by a human evaluator.

以往的动态评估通常采用二元成功率或相对于"真实轨迹"的完成率来评估任务完成度。然而，现实任务往往没有二元成功标志或单一的真实动作序列。例如，"为帕洛阿尔托规划一天行程"这类任务可能涉及探索和信息汇总，存在多种合理方案。因此，我们旨在衡量人类满意度，而非与真实轨迹的精确匹配。对于每个任务，我们首先手动编写一份评分标准列表 (示例见图 5(a))，包含里程碑步骤 (如"打开 Tripadvisor") 和满意度标准 (如"浏览多个景点")。然后定义满意度得分 (SS) 为满足的评分标准数量除以评分标准总数，由人工评估者判断。

We also include Action Accuracy (AA) and Reflection Accuracy (RA) as metrics to evaluate action-level performance. These metrics are also assessed by humans through a review of recorded screenshots and action histories. Finally, we include a Termination Error (TE) rate to reflect the agent's robustness and error recovery capability. There

我们还包括动作准确率 (AA) 和反思准确率 (RA) 作为评估动作层面表现的指标。这些指标同样通过人工审查记录的截图和动作历史进行评估。最后，我们引入终止错误率 (TE) 以反映代理的鲁棒性和错误恢复能力。这里

are five ways an agent can exit from performing a task: (1) self-reported success: the agent decides to stop on its own; (2) reaching the maximum number of iterations: we set the maximum iteration count to 40 to prevent infinite loops; (3) reaching the maximum number of consecutive errors: if the agent has an action reflector and it identifies 3 consecutive errors, the agent is exited; (4) reaching the maximum number of repeated actions: if the agent performs the exact same action (excluding Swipe and Back) more than 3 consecutive times; (5) any other errors, such as errors when parsing the raw response into a valid action. If a task exits in one of the ways described in 2-5, it is marked as having a termination error. The TE rate is computed as the ratio of tasks with termination errors to all tasks.

代理完成任务时有五种退出方式:(1) 自我报告成功: 代理自主决定停止；(2) 达到最大迭代次数: 我们设置最大迭代次数为 40 以防止无限循环；(3) 达到最大连续错误次数: 如果代理具备动作反思器且识别出 3 次连续错误，则退出代理；(4) 达到最大连续重复动作次数: 如果代理连续执行超过 3 次完全相同的动作 (不包括滑动和返回)；(5) 其他错误，如解析原始响应为有效动作时出错。如果任务以第 2 至 5 种方式退出，则标记为终止错误。终止错误率计算为终止错误任务数与所有任务数之比。

3.3. Evaluating Self-Evolving Mobile Agents

3.3. 评估自我进化的移动代理

To the best of our knowledge, this is the first work exploring evaluation in cross-task evolution settings. We consider two variants of Mobile-Agent-E: with and without the self-evolution module. When self-evolution module is enabled-referred to as Mobile-Agent-E + Evo-the agent performs sequentially across tasks within each scenario from the Mobile-Eval-E benchmark. The five tasks in a scenario share a persistent long-term memory. At the end of the k -th task, the Experience Reflectors are prompted to update the long-term memory based on the interaction history of the current task as well as the queries for the remaining $5-k$ tasks. This mimics the implicit requirement for an evolving agent to plan ahead, storing relevant knowledge for future interactions. In this setting, tasks performed later in the sequence benefit from a greater accumulation of Tips and Shortcuts, enabling us to analyze the progressive impact of self-evolution over time (detailed in Figure 6).

据我们所知，这是首个探索跨任务进化设置下评估的工作。我们考虑 Mobile-Agent-E 的两个变体: 带自我进化模块和不带自我进化模块。启用自我进化模块时，称为 Mobile-Agent-E + Evo，代理在 Mobile-Eval-E 基准的每个场景内顺序执行任务。场景中的五个任务共享持久的长期记忆。在第 k 个任务结束时，经验反思器根据当前任务的交互历史及剩余 $5-k$ 个任务的查询提示更新长期记忆。这模拟了进化代理隐含的提前规划需求，存储未来交互相关知识。在此设置下，序列后期执行的任务受益于更多积累的技巧和捷径，使我们能够分析自我进化随时间的渐进影响 (详见图 6)。

3.4. Models

3.4. 模型

Baselines. We compare against a wide range of open-sourced mobile agent frameworks, including AppAgent (Zhang et al., 2023), Mobile-Agent-v1 (Wang et al., 2024b), and Mobile-Agent-v2 (Wang et al., 2024a). To maximize an apple-to-apple comparison with Mobile-Agent-v2, which is the previous state-of-the-art, we apply an identical atomic operation space, perception model, and initial Tips to Mobile-Agent-v2 as Mobile-Agent-E. AppAgent originally requires an additional exploration phase, which does not fit our setting; thus, we add the initial Tips as additional knowledge.

基线。我们对比了多种开源移动代理框架，包括 AppAgent(Zhang 等，2023)、Mobile-Agent-v1(Wang 等，2024b) 和 Mobile-Agent-v2(Wang 等，2024a)。为实现与之前最先进的 Mobile-Agent-v2 的公平比较，我们为 Mobile-Agent-v2 应用了与 Mobile-Agent-E 相同的原子操作空间、感知模型和初始技巧。AppAgent 原本需要额外的探索阶段，不符合我们的设置，因此我们添加了初始技巧作为额外知识。

Backbones. We explore using various large multimodal models (LMM) as backbones for the reasoning agents, including GPT-4o (OpenAI, 2024) [!!] , Claude-3.5-Sonnet (An-

骨干网络。我们探索使用各种大型多模态模型 (LMM) 作为推理代理的骨干网络，包括 GPT-4o(OpenAI，2024) [!!] ，Claude-3.5-Sonnet(An-

---

IGPT-40 version: gpt-40-2024-11-20

IGPT-40 版本:gpt-40-2024-11-20

Table 3. Comparison with state-of-the-art models on the Mobile-Eval-E benchmark, using GPT-40 as the backbone. Mobile-Agent-E outperforms previous SOTA models by a significant margin across all metrics, demonstrating superior long-term planning, decision accuracy, and error recovery. Enabling self-evolution (Mobile-Agent-E + Evo) further enhances performance. Reflection Accuracy for AppAgent and Mobile-Agent-v1 are omitted since they do not have action reflectors.

表 3. 在 Mobile-Eval-E 基准测试中，使用 GPT-40 作为骨干网络与最先进模型的比较。Mobile-Agent-E 在所有指标上均显著优于之前的 SOTA 模型，展现出卓越的长期规划、决策准确性和错误恢复能力。启用自我进化 (Mobile-Agent-E + Evo) 进一步提升了性能。由于 AppAgent 和 Mobile-Agent-v1 没有动作反射器，故省略其反射准确率。

Model	Type	Satisfaction Score (%)↑	Action Accuracy (%) ↑	Reflection Accuracy (%) ↑	Termination Error (%) ↓
AppAgent (Zhang et al., 2023)	Single-Agent	25.2	60.7	-	96.0
Mobile-Agent-v1 (Wang et al., 2024b)	Single-Agent	45.5	69.8	-	68.0
Mobile-Agent-v2 (Wang et al., 2024a)	Multi-Agent	53.0	73.2	96.7	52.0
Mobile-Agent-E	Multi-Agent	75.1	85.9	97.4	32.0
Mobile-Agent-E + Evo	Multi-Agent	86.9	90.4	97.8	12.0

模型	类型	满意度评分 (%)↑	动作准确率 (%) ↑	反思准确率 (%) ↑	终止错误率 (%) ↓
AppAgent (Zhang 等, 2023)	单智能体	25.2	60.7	-	96.0
Mobile-Agent-v1 (Wang 等, 2024b)	单智能体	45.5	69.8	-	68.0
Mobile-Agent-v2 (Wang 等, 2024a)	多智能体	53.0	73.2	96.7	52.0
Mobile-Agent-E	多智能体	75.1	85.9	97.4	32.0
Mobile-Agent-E + 进化	多智能体	86.9	90.4	97.8	12.0

Table 4. Results on different large multimodal model backbones, including GPT-40, Gemini, and Claude. The metrics SS, AA, RA, and TE represent Satisfaction Score, Action Accuracy, Reflection Accuracy, and Termination Error, respectively, expressed as percentages.

表 4. 不同大型多模态模型骨干的结果，包括 GPT-40、Gemini 和 Claude。指标 SS、AA、RA 和 TE 分别表示满意度评分 (Satisfaction Score)、动作准确率 (Action Accuracy)、反思准确率 (Reflection Accuracy) 和终止错误率 (Termination Error)，均以百分比表示。

Model	Gemini-1.5-pro				Claude-3.5-Sonnet				GPT-40			
	SS↑	AA↑	RA↑	TE↓	SS↑	AA↑	RA↑	TE↓	SS↑	AA↑	RA↑	TE↓
Mobile-Agent-v2 (Wang et al., 2024a)	50.8	63.4	83.9	64.0	70.9	76.4	96.9	32.0	53.0	73.2	96.7	52.0
Mobile-Agent-E	70.9	74.3	91.3	48.0	75.5	91.1	99.1	12.0	75.1	85.9	97.4	32.0
Mobile-Agent-E + Evo	71.2	77.4	89.6	48.0	83.0	91.4	99.7	12.0	86.9	90.4	97.8	12.0

模型	Gemini-1.5-pro				Claude-3.5-Sonnet				GPT-40			
	SS↑	AA↑	RA↑	TE↓	SS↑	AA↑	RA↑	TE↓	SS↑	AA↑	RA↑	TE↓
Mobile-Agent-v2 (Wang 等, 2024a)	50.8	63.4	83.9	64.0	70.9	76.4	96.9	32.0	53.0	73.2	96.7	52.0
Mobile-Agent-E	70.9	74.3	91.3	48.0	75.5	91.1	99.1	12.0	75.1	85.9	97.4	32.0
Mobile-Agent-E + Evo	71.2	77.4	89.6	48.0	83.0	91.4	99.7	12.0	86.9	90.4	97.8	12.0

thropic, 2024)**, and Gemini-1.5-pro (Team et al., 2024)**. Unless otherwise specified, the default backbone for all models is GPT-40.

thropic，2024)**，以及 Gemini-1.5-pro(Team 等，2024)**。除非另有说明，所有模型的默认主干网络均为 GPT-40。

Perceptor Implementation in Mobile-Agent-E. We closely follow Mobile-Agent-v2 (Wang et al., 2024a) to implement the Perceptor with slight modifications. We use DBNet#(Liao et al., 2020) and ConvNextViT-document## from ModelScope for OCR detection and recognition respectively. We use GroundingDINO (Liu et al., 2023) for icon grounding and Qwen-VL-Plus (Bai et al., 2023) for generating captions for each cropped icon.

Mobile-Agent-E 中的感知器实现。我们紧密遵循 Mobile-Agent-v2(Wang 等, 2024a) 来实现感知器, 进行了轻微修改。我们分别使用 ModelScope 中的 DBNet#(Liao 等，2020) 和 ConvNextViT-document## 进行 OCR 检测和识别。图标定位采用 GroundingDINO(Liu 等，2023)，每个裁剪图标的描述生成则使用 Qwen-VL-Plus(Bai 等，2023)。

4. Results

4. 结果

4.1. Evaluation on Performance

4.1. 性能评估

Comparison with state-of-the-art. Table 3 presents the results on Mobile-Eval-E using an identical GPT-4o backbone for all baselines and Mobile-Agent-E. Mobile-Agent-E outperforms the previous multi-agent state-of-the-art (SOTA) model (Wang et al., 2024a) by 22.1% in the Satisfaction Score. This comparison particularly highlights the effectiveness of the hierarchy in our multi-agent framework. Our approach also demonstrates superior robustness and error recovery capabilities, as indicated by a signifi-

与最先进技术的比较。表 3 展示了在 Mobile-Eval-E 上的结果, 所有基线模型和 Mobile-Agent-E 均采用相同的 GPT-4o 骨干网络。Mobile-Agent-E 在满意度评分上比之前的多智能体最先进模型 (Wang et al., 2024a) 高出 22.1%。这一比较特别凸显了我们多智能体框架中层级结构的有效性。我们的方法还表现出更强的鲁棒性和错误恢复能力，正如显著的指标所示—

cantly lower termination error rate. Moreover, enabling self-evolution further enhances performance, leading to an improvement of 33.9% against the previous SOTA, underscoring the benefit of learning from experience. In §4.3, we provide further analysis of the impact of the evolution module.

终止错误率显著降低。此外, 启用自我进化功能进一步提升了性能, 相较于之前的最先进技术 (SOTA) 提高了 33.9%，凸显了从经验中学习的优势。在 §4.3 节中，我们对进化模块的影响进行了进一步分析。

Varying reasoning backbones. Table 4 shows the comparison with previous SOTA (Wang et al., 2024a) using various backbone LMMs. We observe consistent improvements on all recent LMMs, including GPT-40, Claude-3.5-Sonnet, and Gemini-1.5-pro, with average absolute gains of 22.1% and 15.6% with and without evolution, respectively. Additionally, the benefits of self-evolution appear to be more pronounced in stronger backbones, such as GPT-4o and Claude.

不同的推理骨干。表 4 展示了使用各种骨干大型多模态模型 (LMMs) 与之前的最新技术水平 (Wang et al., 2024a) 进行的比较。我们观察到在所有近期的 LMMs 上均有持续改进，包括 GPT-4o、Claude-3.5-Sonnet 和 Gemini-1.5-pro，分别在有无自我进化的情况下平均绝对提升为 22.1% 和 15.6%。此外，自我进化的优势在更强大的骨干模型 (如 GPT-4o 和 Claude) 中表现得更为显著。

4.2. Evaluation on Efficiency

4.2. 效率评估

Evaluating the efficiency of mobile agents on complex, potentially open-ended tasks is not straightforward. Merely counting the number of steps is not optimal, as many tasks require exploration. A smaller number of steps reflects a quick exit but may result in insufficient exploration. Intuitively, if an agent achieves higher satisfaction, i.e., fulfills more rubrics, in a smaller number of steps, it is considered more efficient. Thus, we introduce the Satisfaction Score vs Steps (SSS) curve to compare and visualize the efficiency of different agents. To plot the SSS curve, we manually examine the recorded trajectories and track the satisfaction of rubrics after each step. We then plot a poly-line with the step number as the x-axis and the Satisfaction Score as the y-axis. To enable visualization of trajectories with different

评估移动智能体在复杂且可能开放式任务中的效率并非易事。仅仅计数步骤数并不理想，因为许多任务需要探索。较少的步骤数反映了快速退出，但可能导致探索不足。直观上，如果一个智能体在较少的步骤内获得更高的满意度，即完成更多的评分标准 (rubrics)，则被认为更高效。因此，我们引入了满意度得分与步骤数 (Satisfaction Score vs Steps，SSS) 曲线，用以比较和可视化不同智能体的效率。为了绘制 SSS 曲线，我们手动检查记录的轨迹，并在每一步后跟踪评分标准的满足情况。然后以步骤数为横轴，满意度得分为纵轴绘制折线图。为了实现对不同轨迹的可视化

**Claude-3.5 version: claude-3-5-sonnet-20241022

**Claude-3.5 版本:claude-3-5-sonnet-20241022

**Gemini-1.5 version: gemini-1.5-pro-latest (Dec 2024)

**Gemini-1.5 版本:gemini-1.5-pro-latest(2024 年 12 月)

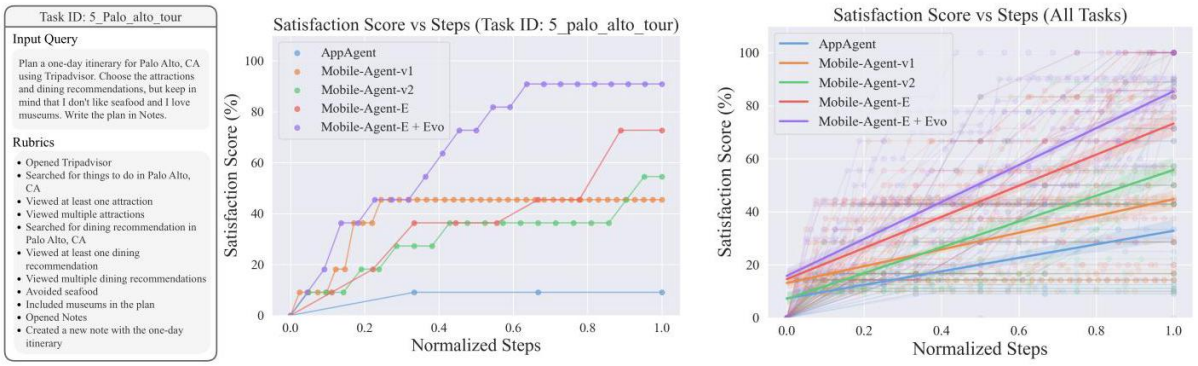
#https://modelscope.cn/models/iic/cv_ resnet18_ocr-detection-db-line-level_damo

#https://modelscope.cn/models/iic/cv_resnet18_ocr-detection-db-line-level_damo

$$https://modelscope.cn/models/iic/cv_ convnextTiny_ocr-recognition-document_damo

$$https://modelscope.cn/models/iic/cv_ convnextTiny_ocr-recognition-document_damo

---



(a) Single Task (task id: 5_palo_alto_tour) (b) All Tasks

Figure 5. Satisfaction Score vs. Steps (SSS) curve for (a) a single task and (b) all tasks. In (a), we also provide a concrete example of the human-written rubrics for the task, which are used to compute the Satisfaction Score during human evaluation. In (b), we additionally include a linear regression line for each model; a steeper and higher line indicates better efficiency for completing the task.

图 5. 满意度评分与步骤数 (SSS) 曲线，分别对应 (a) 单个任务和 (b) 所有任务。在 (a) 中，我们还提供了该任务的人类编写评分标准示例，用于在人类评估中计算满意度评分。在 (b) 中，我们额外为每个模型加入了线性回归线；线条越陡峭且越高，表示完成任务的效率越高。

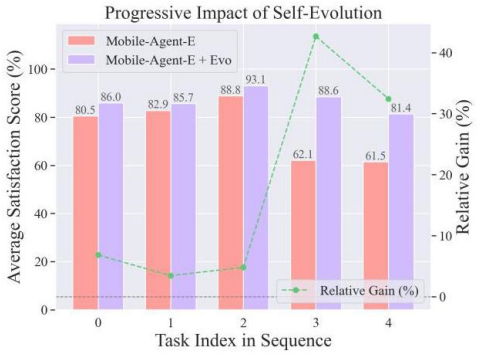


Figure 6. Progressive impact of self-evolution over time. The task index represents the order in which a task is performed in the evolution setting. The results demonstrate that tasks performed later in the sequence show more significant improvements, highlighting the increased benefits from additional iterations of self-evolution.

图 6. 自我进化随时间的渐进影响。任务索引表示任务在进化设置中执行的顺序。结果表明，序列中后期执行的任务表现出更显著的改进，凸显了额外自我进化迭代带来的增益。

lengths on the same graph, we normalize the steps to the range $[0, 1]$. The y-axis of the rightmost point indicates the final satisfaction score. Intuitively, a steeper and higher SSS curve indicates better efficiency and completeness. As shown in Figure 5, we observe that Mobile-Agent-E not only achieves better final performance but also fulfills rubrics faster.

在同一图表上比较长度时，我们将步骤归一化到范围 $[0, 1]$ 。最右侧点的纵轴表示最终满意度得分。直观来看，更陡峭且更高的 SSS 曲线表明更好的效率和完整性。如图 5 所示，我们观察到 Mobile-Agent-E 不仅实现了更优的最终表现，还更快地完成了评分标准。

4.3. Further Analysis

4.3. 进一步分析

Progressive impact of self-evolution over time. The ideal behavior of self-evolution is to progressively bring more benefits to the agent as knowledge accumulates. To investigate this, we group the results of the tasks by their ordering index in each scenario and compare the performance with and without enabling the evolution module. In Figure 6, the x-axis reflects the task index in the sequence

自我进化随时间的渐进影响。自我进化的理想行为是随着知识的积累，逐步为智能体带来更多收益。为探究这一点，我们将各场景中任务按顺序索引分组，并比较启用与未启用进化模块时的性能。在图 6 中，x 轴表示序列中的任务索引

Table 5. Analysis of computational overhead and Shortcut usage. In the inference speed table, the reasoning only section accounts for time spent solely on reasoning agents, while perception + reasoning includes the runtime of the Perceptor on CPU. Shortcut usage statistics are calculated as the ratio of Shortcuts used to the total number of actions performed by the Operator. The use of Shortcuts significantly accelerates inference, achieving comparable times to previous, simpler frameworks.

表 5. 计算开销与快捷方式使用分析。在推理速度表中，仅推理部分指仅智能体推理所花费的时间，而感知 + 推理包括 CPU 上感知器 (Perceptor) 的运行时间。快捷方式使用统计为快捷方式使用次数与操作员执行动作总数的比率。快捷方式的使用显著加快了推理速度，达到了与之前更简单框架相当的时间。

Inference Speed (Seconds per operation)						
Model	Reasoning Only			Perception + Reasoning		
	Gemini	Claude	GPT	Gemini	Claude	GPT
Mobile-Agent-v2	9.8	21.4	12.3	25.6	38.4	43.5
Mobile-Agent-E	16.5	25.5	17.4	30.8	41.0	30.1
Mobile-Agent-E + Evo	12.9	24.8	14.9	27.2	39.6	27.4

推理速度 (每次操作秒数)						
模型	仅推理			感知 + 推理		
	Gemini	Claude	GPT	Gemini	Claude	GPT
Mobile-Agent-v2	9.8	21.4	12.3	25.6	38.4	43.5
Mobile-Agent-E	16.5	25.5	17.4	30.8	41.0	30.1
Mobile-Agent-E + Evo	12.9	24.8	14.9	27.2	39.6	27.4

Shortcut Usage Percentage (%)

快捷方式使用百分比 (%)

Model	Gemini	Claude	GPT
Mobile-Agent-E	11.9	12.8	12.4
Mobile-Agent-E + Evo	14.8	13.2	14.4

模型	双子座 (Gemini)	克劳德 (Claude)	生成式预训练变换器 (GPT)
移动代理-E	11.9	12.8	12.4
移动代理-E + 进化 (Evo)	14.8	13.2	14.4

it is performed, with later tasks having access to Tips and Shortcuts that are updated through more tasks. We observe a generally increasing trend indicating that the gain tends to be more significant in later tasks, demonstrating that the self-evolution module is capable of continuously improving the agent as it experiences more tasks. The gain is not strictly monotonically increasing, as expected, since the difficulty of tasks at different indices varies.

该过程是逐步执行的，后续任务可以访问通过更多任务更新的提示 (Tips) 和捷径 (Shortcuts)。我们观察到总体上呈上升趋势，表明收益在后续任务中更为显著，证明自我进化模块能够随着任务的增加持续提升智能体的能力。收益并非严格单调增加，这是预料之中的，因为不同索引的任务难度存在差异。

Shortcut reduces computational overhead. The hierarchical multi-agent architecture in Mobile-Agent-E significantly improves performance on complex tasks but inevitably increases computational complexity. However, we found that the use of Shortcuts largely mitigates this overhead, enabling Mobile-Agent-E to achieve a speed comparable to that of previous models. In Table 5, we report the

捷径减少了计算开销。Mobile-Agent-E 中的分层多智能体架构显著提升了复杂任务的性能，但不可避免地增加了计算复杂度。然而，我们发现使用捷径在很大程度上缓解了这一开销，使 Mobile-Agent-E 的速度达到与先前模型相当的水平。在表 5 中，我们报告了

Table 6. To investigate the unique impact of Tips, we compute the Satisfaction Score on a subset of instances where no newly generated Shortcuts are used in the trajectory. The results show distinctive benefits from the evolved Tips.

表 6。为了研究提示 (Tips) 的独特影响，我们在轨迹中未使用新生成捷径的实例子集中计算了满意度得分。结果显示进化后的提示带来了显著的益处。

	Gemini	Claude	GPT-4o
Mobile-Agent-E	69.0	75.6	79.7
Mobile-Agent-E + evolved Tips	72.6	85.2	87.5

	双子座	克洛德	GPT-4o
移动代理-E	69.0	75.6	79.7
移动代理-E + 进化提示	72.6	85.2	87.5

seconds per operation averaged across all tasks as well as the usage of Shortcuts. We observe a positive correlation between using more Shortcuts and faster inference speed. This is because a Shortcut enables the execution of multiple operations within a single decision-making iteration. For example, using the Tap_Type_and_Enter Shortcut to perform a search subroutine saves two iterations of perception and reasoning compared to using three atomic actions: Tap, Type, and Enter.

所有任务的平均每次操作耗时以及快捷方式的使用情况。我们观察到使用更多快捷方式与更快的推理速度之间存在正相关关系。这是因为快捷方式使得在单次决策迭代中执行多个操作成为可能。例如，使用 Tap_Type_and_Enter 快捷方式执行搜索子程序，相较于使用三个原子动作: 点击 (Tap)、输入 (Type) 和回车 (Enter)，节省了两次感知和推理的迭代。

Unique impact from Tips. While the impact from Shortcuts is directly visible in the action history, it is less obvious whether the evolved Tips bring distinctive benefits. To visualize this, we filter out task instances where the same set of unique Shortcuts is used or where only atomic actions are employed, and compare the Satisfaction Score with or without the evolution. Table 6 shows that Tips alone serve as an important aspect of self-evolution.

提示 (Tips) 的独特影响。虽然快捷方式的影响可以直接从动作历史中观察到，但进化后的提示是否带来显著益处则不那么明显。为此，我们筛选出使用相同唯一快捷方式集合或仅使用原子动作的任务实例，比较有无进化情况下的满意度评分。表 6 显示，仅提示本身就是自我进化的重要组成部分。

4.4. Case Study: A Closed-Loop Self-Evolving Agent

4.4. 案例研究: 闭环自我进化代理

In real-world mobile usage, after running the agent on a large number of tasks in various scenarios, the accumulated Tips and Shortcuts may grow to an amount where it is no longer feasible to include all of them in the decision-making context. Thus, in this case study, we aim to explore closing the self-evolution loop by introducing two additional Experience Retriever agents for Tips $\mathcal{A}_{ERT}$ and Shortcuts $\mathcal{A}_{ERS}$. We consider a new task in an unknown scenario, as shown in Figure 7. First, we provide all the updated Tips and Shortcuts—after running Mobile-Agent-E on all 5 scenarios (a total of 25 tasks) in Mobile-Eval-E—to the Experience Retrievers. With GPT-4o as the backbone, the updated long-term memory contains a total of 7 unique Shortcuts and 58 Tips, among which 6 Shortcuts and 55 Tips are newly proposed by Mobile-Agent-E during experience reflection. Then, the Experience Retrievers are prompted to select only the relevant Tips and Shortcuts for the current task. The qualitative example in Figure 7 shows that Mobile-Agent-E effectively retrieves and leverages highly relevant Shortcuts and Tips to successfully complete a challenging unseen task. The full list of Tips and Shortcuts after evolution can be found in Appendices G and F.

在真实的移动使用场景中，经过在各种场景下运行代理处理大量任务后，积累的提示和快捷方式可能增多到无法全部纳入决策上下文的程度。因此，在本案例研究中，我们旨在通过引入两个额外的经验检索代理，分别针对提示 $\mathcal{A}_{ERT}$ 和快捷方式 $\mathcal{A}_{ERS}$，探索闭合自我进化循环。我们考虑一个未知场景中的新任务，如图 7 所示。首先，我们将所有更新后的提示和快捷方式——在 Mobile-Eval-E 的 5 个场景 (共 25 个任务) 上运行 Mobile-Agent-E 后获得的——提供给经验检索器。以 GPT-4o 作为基础，更新后的长期记忆包含 7 个独特快捷方式和 58 个提示，其中 6 个快捷方式和 55 个提示是 Mobile-Agent-E 在经验反思过程中新提出的。然后，经验检索器被提示仅选择与当前任务相关的提示和快捷方式。图 7 中的定性示例显示，Mobile-Agent-E 有效检索并利用高度相关的快捷方式和提示，成功完成了一个具有挑战性的未知任务。进化后提示和快捷方式的完整列表见附录 G 和 F。

5. Related Work

5. 相关工作

5.1.GUI Agents

5.1.GUI 代理

The advancement of large multimodal models (LMM) has introduced a new area of agentic research focused on LMM-based GUI agents (Wang et al., 2024c). The goal is to develop AI assistants capable of performing tasks in various GUI environments, such as Web (Deng et al., 2023; Zheng et al., 2024; He et al., 2024; Yoran et al., 2024; Reddy et al., 2024), PC (Hong et al., 2023; Zhang et al., 2024; Liu et al., 2024b; Xie et al., 2024; Tan et al., 2024), and mobile devices (Wang et al., 2024b; Zhang et al., 2023; Li et al., 2024; Wang et al., 2024a; Liu et al., 2024a). In the mobile environment, one line of research focuses on improving the perception and reasoning abilities of a single agent through tool usage (Wang et al., 2024b) and an additional exploration phase (Zhang et al., 2023; Li et al., 2024). Recent studies (Rawles et al., 2024; Wang et al., 2024a) show significant promise by incorporating multiple agents for decision-making and reflection. However, current multi-agent frameworks still face challenges such as short-sighted planning and poor error recovery. Specifically, the ”planning” module in Mobile-Agent-v2 (Wang et al., 2024a) functions primarily as a progress tracker, while the ”decision-making” module continues to handle both high-level planning (e.g., ”what to do next”) and low-level action decisions (e.g., ”where to tap”). A key difference in Mobile-Agent-E is the introduction of a hierarchy among the agents, enabling more effective long-horizon planning and improved low-level action accuracy.

大型多模态模型 (LMM) 的进展引入了基于 LMM 的 GUI 代理 (agentic research) 的新研究领域 (Wang et al., 2024c)。目标是开发能够在各种 GUI 环境中执行任务的 AI 助手，如网页 (Deng et al., 2023；Zheng et al., 2024；He et al., 2024；Yoran et al., 2024；Reddy et al., 2024)、PC(Hong et al., 2023；Zhang et al., 2024；Liu et al., 2024b；Xie et al., 2024；Tan et al., 2024) 和移动设备 (Wang et al., 2024b；Zhang et al., 2023；Li et al., 2024；Wang et al., 2024a；Liu et al., 2024a)。在移动环境中，一类研究聚焦于通过工具使用 (Wang et al., 2024b) 和额外探索阶段 (Zhang et al., 2023；Li et al., 2024) 提升单一代理的感知和推理能力。近期研究 (Rawles et al., 2024；Wang et al., 2024a) 通过引入多代理进行决策和反思展现出显著潜力。然而，当前多代理框架仍面临短视规划和错误恢复能力差等挑战。具体而言，Mobile-Agent-v2(Wang et al., 2024a) 中的“规划”模块主要作为进度跟踪器，而“决策”模块仍负责高层规划 (如“下一步做什么”) 和低层动作决策 (如“点击哪里”)。Mobile-Agent-E 的关键区别在于引入了代理层级结构，实现了更有效的长远规划和更精准的低层动作。

5.2. Self-Evolution in Foundation Models

5.2. 基础模型中的自我进化

Investigating how to make large language models and multimodal models self-improve has long been an active area of research (Tao et al., 2024). One line of work focuses on enhancing the base abilities of foundation models, such as improving reasoning and reducing knowledge hallucination. This includes approaches like iterative refinement (Madaan et al., 2024), self-reflection (Shinn et al., 2024), self-training (Huang et al., 2022), self-improvement (Wang et al., 2024d), and multi-persona collaboration (Wang et al., 2023). Another line of work explores improving task-solving with foundation models through tool learning and tool creation (Cai et al., 2023; Qian et al., 2023; Yuan et al., 2023). In the context of GUI agents, self-evolution is less studied. The skill curation mechanism in Cradle (Tan et al., 2024) shows initial promise in the PC environment; however, no previous work has systematically explored self-evolution in mobile environments. In this work, we demonstrate the importance of self-evolution in both Tips and Shortcuts. Notably, unlike the ”skills” in Cradle, which are directly added to the atomic operation space, we explicitly define preconditions for our Shortcuts, as this is critical for decision-making across multiple apps and varying layouts.

如何使大型语言模型和多模态模型实现自我提升一直是一个活跃的研究领域 (Tao 等，2024)。一类研究工作聚焦于增强基础模型的基本能力，如提升推理能力和减少知识幻觉。这包括迭代优化 (Madaan 等，2024)、自我反思 (Shinn 等，2024)、自我训练 (Huang 等，2022)、自我改进 (Wang 等，2024d) 以及多角色协作 (Wang 等，2023) 等方法。另一类工作则通过工具学习和工具创造 (Cai 等，2023；Qian 等，2023；Yuan 等，2023) 来提升基础模型的任务解决能力。在图形用户界面 (GUI) 代理的背景下，自我进化的研究较少。Cradle(Tan 等，2024) 中的技能策划机制在 PC 环境中展现了初步的潜力；然而，尚无研究系统地探讨移动环境中的自我进化。在本研究中，我们展示了自我进化在提示 (Tips) 和快捷方式 (Shortcuts) 中的重要性。值得注意的是，与 Cradle 中直接添加到原子操作空间的“技能”不同，我们明确为快捷方式定义了前置条件，因为这对于跨多个应用和不同布局的决策至关重要。

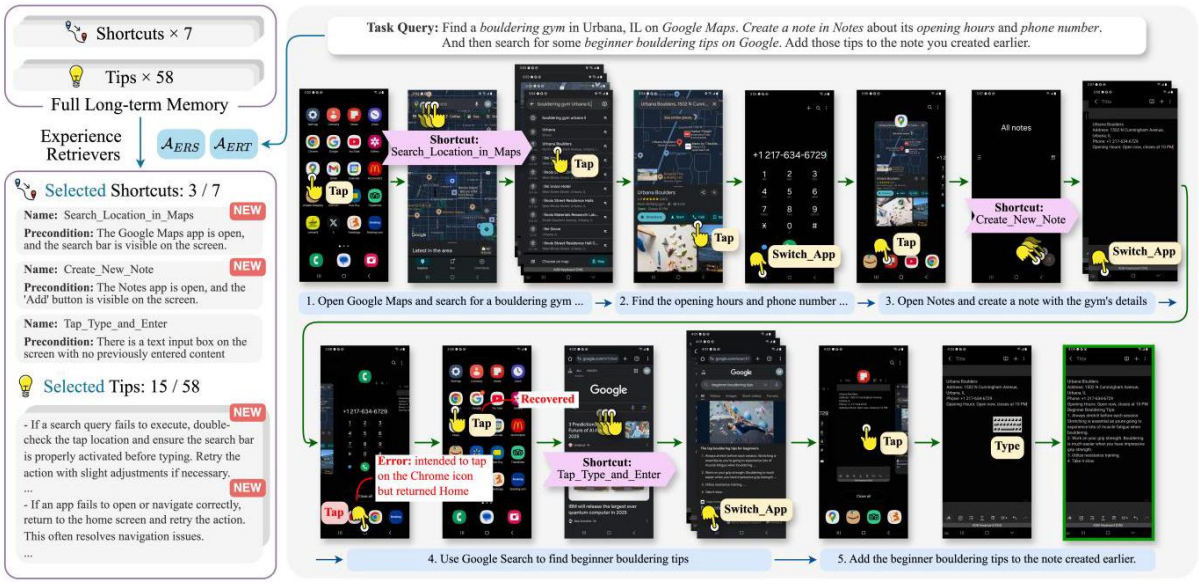


Figure 7. Case study example where relevant Shortcuts and Tips are automatically retrieved from the previously evolved long-term memory and subsequently leveraged to complete an unseen, challenging task. The action trajectory also includes an example where the agent recovers from an error.

图 7。案例研究示例，相关的快捷方式和提示从先前进化的长期记忆中自动检索，并随后被利用以完成一个未见过的挑战性任务。动作轨迹还包括代理从错误中恢复的示例。

6. Conclusion and Future Work

6. 结论与未来工作

We introduce Mobile-Agent-E, a novel mobile assistant featuring a hierarchical multi-agent framework and a self-evolution module that significantly enhances long-term planning, error recovery, and efficiency, excelling in a wide variety of complex real-world tasks. Remaining limitations include the incorrect usage of Shortcuts with invalid preconditions and erroneous agent-generated Shortcuts, with detailed examples provided in Appendix C. Future work will focus on developing improved strategies for generating, invoking, and revising Shortcuts, enhancing personalization to better adapt to individual user needs, and strengthening safety precautions to enable more effective human-agent collaboration.

我们提出了 Mobile-Agent-E，一种新颖的移动助手，具备分层多代理框架和自我进化模块，显著提升了长期规划、错误恢复和效率，在各种复杂的现实任务中表现出色。现存的局限包括快捷方式在前置条件无效时的错误使用以及代理生成的快捷方式存在错误，详细示例见附录 C。未来工作将聚焦于开发更优的快捷方式生成、调用和修订策略，增强个性化以更好地适应用户需求，并强化安全防护措施，以实现更有效的人机协作。

Impact Statement

影响声明

This paper aims to advance the field of LMM-based agents by developing a hierarchical multi-agent framework and benchmark to improve the usability and efficiency of smart-phones in complex, multi-step tasks. While the primary goal is to enhance human-device interaction, the proposed system has the potential for broader societal benefits, particularly in improving accessibility for individuals with disabilities or limited mobility. By enabling more intuitive and automated task management on mobile devices, this framework can assist users with physical impairments, cognitive chal-

本文旨在通过开发分层多代理框架和基准，推动基于大型多模态模型 (LMM) 的代理领域发展，以提升智能手机在复杂多步骤任务中的可用性和效率。虽然主要目标是增强人机交互，但所提系统在社会层面具有更广泛的潜在益处，特别是在改善残障人士或行动不便者的无障碍访问方面。通过实现移动设备上更直观和自动化的任务管理，该框架能够帮助身体障碍、认知挑战或难以精确操作触摸屏的用户，

lenges, or conditions that make precise interactions with touchscreens difficult.

克服操作困难。

While the primary aim is to enhance mobile task efficiency and user accessibility, the development of mobile agents capable of autonomous decision-making introduces potential risks. For example, unauthorized or unintended actions by the agent, such as the misuse of sensitive information including credit card details or private data, could result in serious consequences for users. These risks emphasize the critical need for robust safeguards, error recovery mechanisms, and fail-safe systems to ensure that the agent's actions consistently align with user intentions.

尽管主要目的是提升移动任务效率和用户无障碍性，具备自主决策能力的移动代理的发展也带来了潜在风险。例如，代理可能执行未经授权或非预期的操作，如滥用信用卡信息或私人数据，可能对用户造成严重后果。这些风险凸显了建立强健的安全保障、错误恢复机制和故障保护系统的关键性，以确保代理行为始终符合用户意图。

We are actively pursuing future work that focuses on designing and integrating robust privacy and safety mechanisms. These include explicit user consent workflows for sensitive operations, encryption protocols to protect user data during processing and storage, and automated systems to flag potentially harmful or unauthorized actions. These advancements will be crucial for maximizing the societal benefits of these systems, minimizing potential risks, and building user trust in autonomous mobile agents.

我们正积极开展未来工作，重点设计和集成强有力的隐私与安全机制，包括针对敏感操作的明确用户同意流程、保护用户数据处理和存储的加密协议，以及自动标记潜在有害或未授权操作的系统。这些进展对于最大化系统的社会效益、最小化潜在风险以及建立用户对自主移动代理的信任至关重要。

References

参考文献

Anderson, J. R. Acquisition of cognitive skill. Psychological review, 89(4):369, 1982.

Anderson, J. R. 认知技能的习得。心理学评论，89(4):369，1982。

Anthropic. Claude 3.5 Sonnet, 2024. URL https://www.anthropic.com/news/ 3-5-models-and-computer-use.

Anthropic. Claude 3.5 Sonnet,2024。网址 https://www.anthropic.com/news/3-5-models-and-computer-use。

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-VL: A Frontier Large Vision-Language Model with Versatile Abilities. arXiv preprint arXiv:2308.12966, 2023. URL https://doi.org/10.48550/arXiv.2308.12966.

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., 和 Zhou, J. Qwen-VL: 一款具备多功能能力的前沿大型视觉-语言模型。arXiv 预印本 arXiv:2308.12966，2023。网址 https://doi.org/10.48550/arXiv.2308.12966。

Cai, T., Wang, X., Ma, T., Chen, X., and Zhou, D. Large language models as tool makers. arXiv preprint arXiv:2305.17126, 2023.

Cai, T., Wang, X., Ma, T., Chen, X., 和 Zhou, D. 大型语言模型作为工具制造者。arXiv 预印本 arXiv:2305.17126，2023。

Deng, X., Gu, Y., Zheng, B., Chen, S., Stevens, S., Wang, B., Sun, H., and Su, Y. Mind2web: Towards a generalist agent for the web. In Thirty-seventh Conference on Neural Information Processing Systems, 2023. URL https: //openreview.net/forum?id=kiYqbO3wqw.

Deng, X., Gu, Y., Zheng, B., Chen, S., Stevens, S., Wang, B., Sun, H., 和 Su, Y. Mind2web: 面向通用网络代理的研究。在第三十七届神经信息处理系统会议，2023 年。网址 https://openreview.net/forum?id=kiYqbO3wqw。

He, H., Yao, W., Ma, K., Yu, W., Dai, Y., Zhang, H., Lan, Z., and Yu, D. Webvoyager: Building an end-to-end web agent with large multimodal models. arXiv preprint arXiv:2401.13919, 2024.

He, H., Yao, W., Ma, K., Yu, W., Dai, Y., Zhang, H., Lan, Z., 和 Yu, D. Webvoyager: 基于大型多模态模型构建端到端网络代理。arXiv 预印本 arXiv:2401.13919，2024 年。

Hong, W., Wang, W., Lv, Q., Xu, J., Yu, W., Ji, J., Wang, Y., Wang, Z., Dong, Y., Ding, M., and Tang, J. Cogagent: A visual language model for gui agents, 2023.

Hong, W., Wang, W., Lv, Q., Xu, J., Yu, W., Ji, J., Wang, Y., Wang, Z., Dong, Y., Ding, M., 和 Tang, J. Cogagent: 面向图形用户界面代理的视觉语言模型，2023 年。

Huang, J., Gu, S. S., Hou, L., Wu, Y., Wang, X., Yu, H., and Han, J. Large language models can self-improve. arXiv preprint arXiv:2210.11610, 2022.

Huang, J., Gu, S. S., Hou, L., Wu, Y., Wang, X., Yu, H., 和 Han, J. 大型语言模型能够自我提升。arXiv 预印本 arXiv:2210.11610，2022 年。

Li, Y., Zhang, C., Yang, W., Fu, B., Cheng, P., Chen, X., Chen, L., and Wei, Y. Appagent v2: Advanced agent for flexible mobile interactions. arXiv preprint arXiv:2408.11824, 2024.

Li, Y., Zhang, C., Yang, W., Fu, B., Cheng, P., Chen, X., Chen, L., 和 Wei, Y. Appagent v2: 用于灵活移动交互的高级代理。arXiv 预印本 arXiv:2408.11824，2024 年。

Liao, M., Wan, Z., Yao, C., Chen, K., and Bai, X. Real-time scene text detection with differentiable binarization. In Proceedings of the AAAI conference on artificial intelligence, volume 34, pp. 11474-11481, 2020.

Liao, M., Wan, Z., Yao, C., Chen, K., 和 Bai, X. 基于可微分二值化的实时场景文本检测。发表于人工智能协会会议论文集，卷 34，第 11474-11481 页，2020 年。

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., and Zhang, L. Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. CoRR, abs/2303.05499, 2023. doi: 10.48550/ARXIV.2303.05499. URL https:// doi.org/10.48550/arXiv.2303.05499.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., 和 Zhang, L. Grounding DINO: 将 DINO 与基于定位的预训练结合用于开放集目标检测。CoRR, abs/2303.05499, 2023 年。doi: 10.48550/ARXIV.2303.05499。网址 https://doi.org/10.48550/arXiv.2303.05499。

Liu, X., Qin, B., Liang, D., Dong, G., Lai, H., Zhang, H., Zhao, H., Iong, I. L., Sun, J., Wang, J., et al. Autoglm: Autonomous foundation agents for guis. arXiv preprint arXiv:2411.00820, 2024a.

Liu, X., Qin, B., Liang, D., Dong, G., Lai, H., Zhang, H., Zhao, H., Iong, I. L., Sun, J., Wang, J., 等。Autoglm: 面向图形用户界面的自主基础代理。arXiv 预印本 arXiv:2411.00820，2024a。

Liu, X., Zhang, T., Gu, Y., Iong, I. L., Xu, Y., Song, X., Zhang, S., Lai, H., Liu, X., Zhao, H., et al. Visualagentbench: Towards large multimodal models as visual foundation agents. arXiv preprint arXiv:2408.06327, 2024b.

Liu, X., Zhang, T., Gu, Y., Iong, I. L., Xu, Y., Song, X., Zhang, S., Lai, H., Liu, X., Zhao, H., 等。Visualagentbench: 迈向大型多模态模型作为视觉基础代理。arXiv 预印本 arXiv:2408.06327，2024b。

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., et al. Self-refine: Iterative refinement with self-feedback. Advances in Neural Information Processing Systems, 36, 2024.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., 等。Self-refine: 基于自我反馈的迭代精炼。神经信息处理系统进展，36 卷，2024 年。

Nguyen, D., Chen, J., Wang, Y., Wu, G., Park, N., Hu, Z., Lyu, H., Wu, J., Aponte, R., Xia, Y., et al. Gui agents: A survey. arXiv preprint arXiv:2412.13501, 2024.

Nguyen, D., Chen, J., Wang, Y., Wu, G., Park, N., Hu, Z., Lyu, H., Wu, J., Aponte, R., Xia, Y., 等。图形用户界面代理: 综述。arXiv 预印本 arXiv:2412.13501，2024 年。

OpenAI. GPT-40 System Card, 2024. URL https: //cdn.openai.com/gpt-40-system-card. pdf.

OpenAI. GPT-40 系统说明，2024 年。网址 https://cdn.openai.com/gpt-40-system-card.pdf。

Qian, C., Han, C., Fung, Y. R., Qin, Y., Liu, Z., and Ji, H. Creator: Tool creation for disentangling abstract and concrete reasoning of large language models. arXiv preprint arXiv:2305.14318, 2023.

Qian, C., Han, C., Fung, Y. R., Qin, Y., Liu, Z., 和 Ji, H. Creator: 用于解耦大型语言模型抽象与具体推理的工具创建。arXiv 预印本 arXiv:2305.14318，2023 年。

Rawles, C., Clinckemaillie, S., Chang, Y., Waltz, J., Lau, G., Fair, M., Li, A., Bishop, W., Li, W., Campbell-Ajala, F., et al. Androidworld: A dynamic benchmarking environment for autonomous agents. arXiv preprint arXiv:2405.14573, 2024.

Rawles, C., Clinckemaillie, S., Chang, Y., Waltz, J., Lau, G., Fair, M., Li, A., Bishop, W., Li, W., Campbell-Ajala, F., 等。Androidworld: 面向自主代理的动态基准环境。arXiv 预印本 arXiv:2405.14573，2024 年。

Reddy, R. G., Mukherjee, S., Kim, J., Wang, Z., Hakkani-Tur, D., and Ji, H. Infogent: An agent-based framework for web information aggregation. arXiv preprint arXiv:2410.19054, 2024.

Reddy, R. G., Mukherjee, S., Kim, J., Wang, Z., Hakkani-Tur, D., 和 Ji, H. Infogent: 一个基于代理的网页信息聚合框架。arXiv 预印本 arXiv:2410.19054, 2024。

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems, 36, 2024.

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., 和 Yao, S. Reflexion: 具备语言强化学习的语言代理。神经信息处理系统进展，第 36 卷，2024。

Squire, L. R. and Zola, S. M. Structure and function of declarative and nondeclarative memory systems. Proceedings of the National Academy of Sciences, 93(24): 13515-13522, 1996.

Squire, L. R. 和 Zola, S. M. 陈述性记忆和非陈述性记忆系统的结构与功能。美国国家科学院院刊，93(24): 13515-13522, 1996。

Tan, W., Ding, Z., Zhang, W., Li, B., Zhou, B., Yue, J., Xia, H., Jiang, J., Zheng, L., Xu, X., et al. Towards general computer control: A multimodal agent for red dead redemption ii as a case study. In ICLR 2024 Workshop on Large Language Model (LLM) Agents, 2024.

Tan, W., Ding, Z., Zhang, W., Li, B., Zhou, B., Yue, J., Xia, H., Jiang, J., Zheng, L., Xu, X., 等. 迈向通用计算机控制: 以《荒野大镖客 2》为案例的多模态代理。ICLR 2024 大型语言模型 (LLM) 代理研讨会，2024。

Tao, Z., Lin, T.-E., Chen, X., Li, H., Wu, Y., Li, Y., Jin, Z., Huang, F., Tao, D., and Zhou, J. A survey on self-evolution of large language models. arXiv preprint arXiv:2404.14387, 2024.

Tao, Z., Lin, T.-E., Chen, X., Li, H., Wu, Y., Li, Y., Jin, Z., Huang, F., Tao, D., 和 Zhou, J. 大型语言模型自我进化综述。arXiv 预印本 arXiv:2404.14387, 2024。

Team, G., Georgiev, P., Lei, V. I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S.,

Team, G., Georgiev, P., Lei, V. I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S.,

et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv preprint arXiv:2403.05530, 2024.

等. Gemini 1.5: 解锁跨百万上下文标记的多模态理解。arXiv 预印本 arXiv:2403.05530, 2024。

Tulving, E. Episodic memory: From mind to brain. Annual review of psychology, 53(1):1-25, 2002.

Tulving, E. 情景记忆: 从心智到大脑。心理学年鉴，53(1):1-25, 2002。

Wang, J., Xu, H., Jia, H., Zhang, X., Yan, M., Shen, W., Zhang, J., Huang, F., and Sang, J. Mobile-agent-v2: Mobile device operation assistant with effective navigation via multi-agent collaboration. arXiv preprint arXiv:2406.01014, 2024a.

Wang, J., Xu, H., Jia, H., Zhang, X., Yan, M., Shen, W., Zhang, J., Huang, F., 和 Sang, J. Mobile-agent-v2: 通过多代理协作实现高效导航的移动设备操作助手。arXiv 预印本 arXiv:2406.01014, 2024a。

Wang, J., Xu, H., Ye, J., Yan, M., Shen, W., Zhang, J., Huang, F., and Sang, J. Mobile-agent: Autonomous multi-modal mobile device agent with visual perception. arXiv preprint arXiv:2401.16158, 2024b.

Wang, J., Xu, H., Ye, J., Yan, M., Shen, W., Zhang, J., Huang, F., 和 Sang, J. Mobile-agent: 具备视觉感知的自主多模态移动设备代理。arXiv 预印本 arXiv:2401.16158, 2024b。

Wang, S., Liu, W., Chen, J., Gan, W., Zeng, X., Yu, S., Hao, X., Shao, K., Wang, Y., and Tang, R. Gui agents with foundation models: A comprehensive survey. arXiv preprint arXiv:2411.04890, 2024c.

Wang, S., Liu, W., Chen, J., Gan, W., Zeng, X., Yu, S., Hao, X., Shao, K., Wang, Y., 和 Tang, R. 基于基础模型的 GUI 代理: 综合综述。arXiv 预印本 arXiv:2411.04890, 2024c。

Wang, Z., Mao, S., Wu, W., Ge, T., Wei, F., and Ji, H. Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. arXiv preprint arXiv:2307.05300, 2023.

Wang, Z., Mao, S., Wu, W., Ge, T., Wei, F., 和 Ji, H. 释放大型语言模型中的新兴认知协同效应: 通过多角色自我协作实现任务解决代理。arXiv 预印本 arXiv:2307.05300, 2023。

Wang, Z., Hou, L., Lu, T., Wu, Y., Li, Y., Yu, H., and Ji, H. Enable lanuguage models to implicitly learn self-improvement from data. In Proc. The Twelfth International Conference on Learning Representations (ICLR2024),

2024d.

Wang, Z., Hou, L., Lu, T., Wu, Y., Li, Y., Yu, H., 和 Ji, H. 使语言模型能够从数据中隐式学习自我提升。第十二届国际学习表征会议 (ICLR2024) 论文集，2024d。

Xie, T., Zhang, D., Chen, J., Li, X., Zhao, S., Cao, R., Hua, T. J., Cheng, Z., Shin, D., Lei, F., et al. Os-world: Benchmarking multimodal agents for open-ended tasks in real computer environments. arXiv preprint arXiv:2404.07972, 2024.

Xie, T., Zhang, D., Chen, J., Li, X., Zhao, S., Cao, R., Hua, T. J., Cheng, Z., Shin, D., Lei, F., 等. Os-world: 在真实计算机环境中对多模态代理进行开放式任务基准测试。arXiv 预印本 arXiv:2404.07972, 2024。

Yoran, O., Amouyal, S. J., Malaviya, C., Bogin, B., Press, O., and Berant, J. Assistantbench: Can web agents solve realistic and time-consuming tasks?, 2024. URL https: //arxiv.org/abs/2407.15711.

Yoran, O., Amouyal, S. J., Malaviya, C., Bogin, B., Press, O., 和 Berant, J. Assistantbench: 网络代理能否解决现实且耗时的任务？, 2024. URL https: //arxiv.org/abs/2407.15711。

Yuan, L., Chen, Y., Wang, X., Fung, Y. R., Peng, H., and Ji, H. Craft: Customizing llms by creating and retrieving from specialized toolsets. arXiv preprint arXiv:2309.17428, 2023.

Yuan, L., Chen, Y., Wang, X., Fung, Y. R., Peng, H., 和 Ji, H. Craft: 通过创建和检索专用工具集定制大型语言模型。arXiv 预印本 arXiv:2309.17428, 2023。

Zhang, C., Yang, Z., Liu, J., Han, Y., Chen, X., Huang, Z., Fu, B., and Yu, G. Appagent: Multimodal agents as smartphone users, 2023.

Zhang, C., Yang, Z., Liu, J., Han, Y., Chen, X., Huang, Z., Fu, B., 和 Yu, G. Appagent: 作为智能手机用户的多模态代理，2023 年。

Zhang, C., Li, L., He, S., Zhang, X., Qiao, B., Qin, S., Ma, M., Kang, Y., Lin, Q., Rajmohan, S., Zhang, D., and Zhang, Q. UFO: A UI-Focused Agent for Windows OS Interaction. arXiv preprint arXiv:2402.07939, 2024.

Zhang, C., Li, L., He, S., Zhang, X., Qiao, B., Qin, S., Ma, M., Kang, Y., Lin, Q., Rajmohan, S., Zhang, D., 和 Zhang, Q. UFO: 面向 Windows 操作系统交互的 UI 聚焦代理。arXiv 预印本 arXiv:2402.07939，2024 年。

Zheng, B., Gou, B., Kil, J., Sun, H., and Su, Y. Gpt- 4v(ision) is a generalist web agent, if grounded. In Forty-first International Conference on Machine Learning, 2024. URL https://openreview.net/forum? id=piecKJ2D1B.

Zheng, B., Gou, B., Kil, J., Sun, H., 和 Su, Y. Gpt-4v(ision) 是一个通用的网络代理，前提是有落地支持。发表于第 41 届国际机器学习大会，2024 年。网址 https://openreview.net/forum?id=piecKJ2D1B。

A. Full Trajectory Comparison Example with Previous SOTA

A. 与之前最先进方法的完整轨迹对比示例

Figure 8 presents the full trajectory of the task shown in Figure 1, comparing the previous state-of-the-art, Mobile-Agent-v2 (Wang et al., 2024a), and our proposed Mobile-Agent-E. Mobile-Agent-v2 suffers from early termination after interacting with two Apps, whereas Mobile-Agent-E fulfills all rubrics and stops at the App offering the best deal.

图 8 展示了图 1 中任务的完整轨迹，比较了之前的最先进方法 Mobile-Agent-v2(Wang 等，2024a) 与我们提出的 Mobile-Agent-E。Mobile-Agent-v2 在与两个应用交互后早期终止，而 Mobile-Agent-E 完成了所有评分标准，并在提供最佳优惠的应用处停止。

B. Error Recovery with Escalation to Manager

B. 通过升级至管理者的错误恢复

Figure 9 illustrates how the error escalation mechanism in Mobile-Agent-E enhances error recovery ability. A detailed description of the example is provided in the caption.

图 9 说明了 Mobile-Agent-E 中的错误升级机制如何增强错误恢复能力。示例的详细描述见图注。

C. Remaining Limitations

C. 现存的局限性

C.1. Misuse of Shortcuts due to Incorrect Perception of Phone State

C.1. 由于对手机状态感知错误导致快捷方式误用

Although we explicitly require the Operator to verify the current phone state to ensure it fulfills the precondition of a Shortcut before calling it, there are still cases where the model incorrectly perceives the state, resulting in the misuse of Shortcuts in an invalid state. Figure 10 illustrates an example of such error. A detailed description of the example is provided in the caption. This type of error could potentially be mitigated by employing a dedicated agent for verifying preconditions or by enhancing the perception module to better understand phone states.

尽管我们明确要求操作员验证当前手机状态以确保满足调用快捷方式的前提条件，但仍存在模型错误感知状态，导致在无效状态下误用快捷方式的情况。图 10 展示了此类错误的示例。示例的详细描述见图注。此类错误有望通过使用专门的代理验证前提条件或增强感知模块以更好理解手机状态来减轻。

C.2. Errors and Imperfections in Self-Evolved Shortcuts

C.2. 自我进化快捷方式中的错误与不完善

Although effective in most cases, we still observe errors and imperfections in the agent-generated Shortcuts during self-evolution. These issues can lead to propagated errors when an erroneous Shortcut is used in subsequent tasks. Figure 11 illustrates an example of such erroneous and imperfect Shortcuts. A detailed description of the example is provided in the caption. This highlights the need for future work on approaches to generate higher-quality Shortcuts and equipping the agent with the ability to reflect on and revise generated Shortcuts in subsequent tasks.

尽管在大多数情况下有效，我们仍观察到代理生成的快捷方式在自我进化过程中存在错误和不完善。这些问题可能导致错误快捷方式在后续任务中传播错误。图 11 展示了此类错误和不完善快捷方式的示例。示例的详细描述见图注。这凸显了未来工作中生成更高质量快捷方式的方法以及赋予代理在后续任务中反思和修正生成快捷方式能力的必要性。

D.All Tasks in Mobile-Eval-E Benchmark

D. Mobile-Eval-E 基准中的所有任务

Table 7 presents the input queries, involved App types, and scenarios for all Mobile-Eval-E tasks. The complete list of rubrics and human reference operation sequences is provided in the supplementary material.

表 7 展示了所有 Mobile-Eval-E 任务的输入查询、涉及的应用类型和场景。完整的评分标准和人工参考操作序列见补充材料。

E. Atomic Operation Space

E. 原子操作空间

Table 8 presents all atomic operations considered in Mobile-Agent-E.

表 8 展示了 Mobile-Agent-E 中考虑的所有原子操作。

F. Full list of Self-Evolved Shortcuts

F. 自我进化快捷方式完整列表

Figure 12 shows a full list of generated Shortcuts by Mobile-Agent-E after self-evolution on all 25 tasks from Mobile-Eval-E benchmark.

图 12 展示了 Mobile-Agent-E 在 Mobile-Eval-E 基准的全部 25 个任务上自我进化后生成的快捷方式完整列表。

G. Full list of Self-Evolved Tips

G. 自我进化提示完整列表

Figure 13 shows a full list of generated Tips by Mobile-Agent-E after self-evolution on all 25 tasks from Mobile-Eval-E benchmark.

图 13 展示了 Mobile-Agent-E 在 Mobile-Eval-E 基准的全部 25 个任务上自我进化后生成的提示完整列表。

Figure 8. Full trajectory comparison between the previous state-of-the-art, Mobile-Agent-v2 (Wang et al., 2024a), and Mobile-Agent-E.

图 8. 先前最先进方法 Mobile-Agent-v2(Wang 等，2024a) 与 Mobile-Agent-E 的完整轨迹对比。

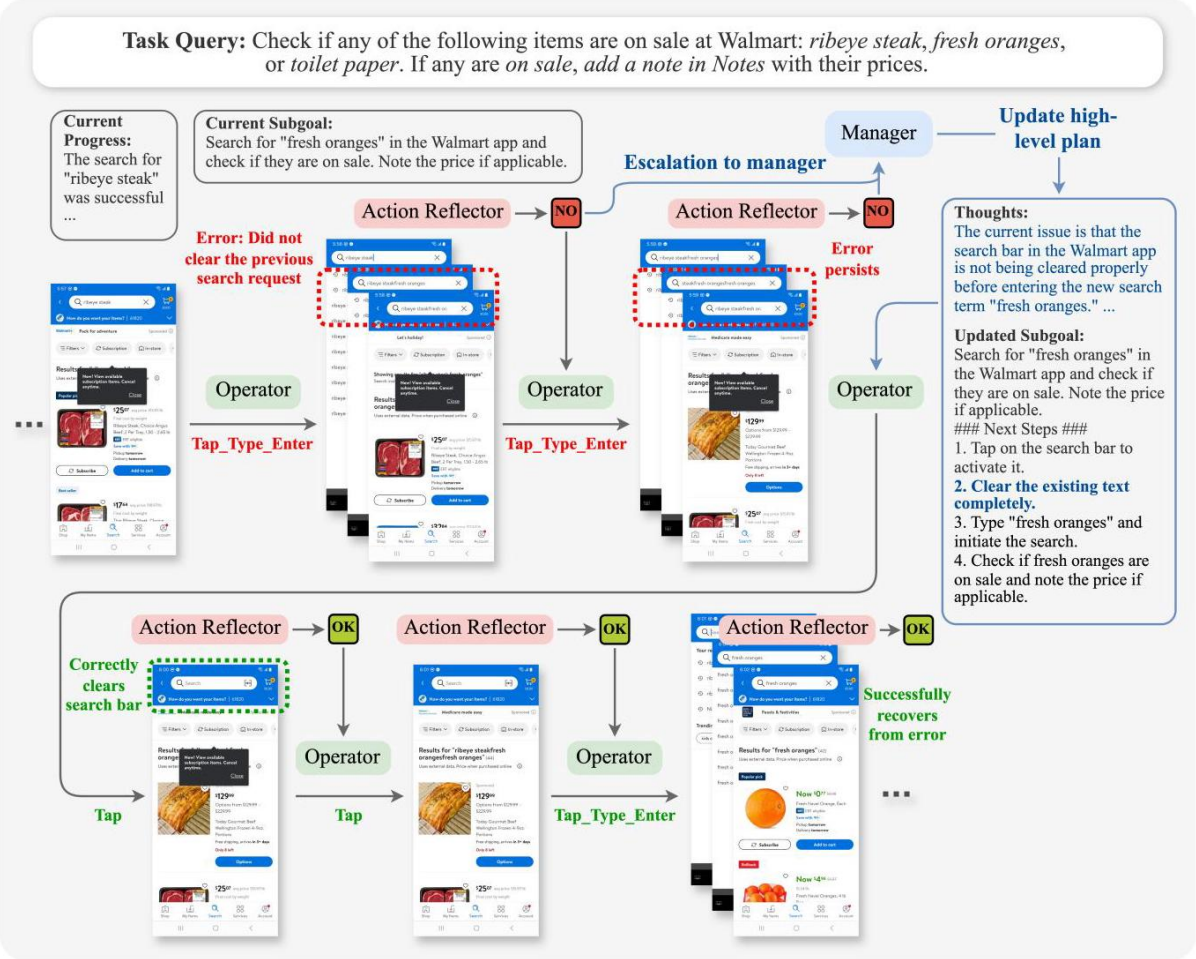


Figure 9. Error recovery with escalation. The task requires the agent to search for three different items on Walmart and note their sales information. At the step shown in the figure, the agent has already searched for ribeye steak and intends to search for fresh oranges next. However, the Operator erroneously calls the Shortcut that inputs text into the search bar and performs a search without clearing the previously entered text. Although the Action Reflector raises an error, the subgoal remains unchanged, and the Operator fails to rectify the error on the second attempt. After observing two consecutive errors, the error is escalated to the Manager, which correctly identifies the problem and revises the subgoal with detailed, decomposed steps to address the error. This helps the Operator correctly recover from the previous error by first tapping the ” × ” icon to clear the previous search query.

图 9. 带升级的错误恢复。该任务要求代理在沃尔玛网站搜索三种不同商品并记录其销售信息。图中所示步骤，代理已搜索肋眼牛排，接下来准备搜索新鲜橙子。然而，操作器错误地调用了一个快捷方式，该快捷方式在搜索栏输入文本并执行搜索，但未清除之前输入的文本。尽管动作反射器 (Action Reflector) 报错，子目标未变，操作器第二次尝试未能纠正错误。观察到连续两次错误后，错误被升级至管理者 (Manager)，管理者正确识别问题并通过详细分解步骤修正子目标，帮助操作器先点击“×”图标清除之前的搜索查询，从而正确恢复之前的错误。

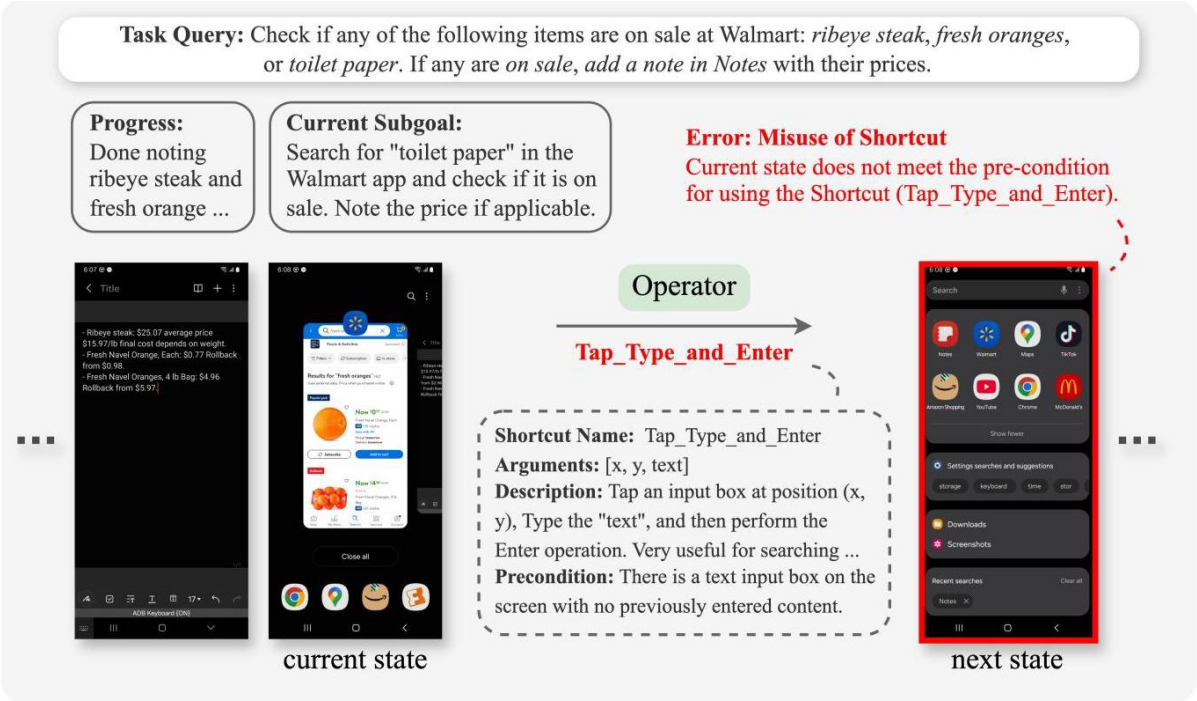


Figure 10. Example of misuse of Shortcuts in an invalid state. At the current step, as shown in the figure, the agent intended to switch back to Walmart to search for the final item requested by the user. While it correctly performs the ”Switch_App” operation, it then calls a Shortcut for searching without realizing that it is not yet in the App where the search bar is available.

图 10. 在无效状态下误用快捷方式示例。当前步骤如图所示，代理意图切换回沃尔玛搜索用户请求的最后一件商品。虽然正确执行了“Switch_App”操作，但随后调用了搜索快捷方式，却未意识到尚未进入带有搜索栏的应用。

{

”name”: ”Search_Location_in_Maps”,

”name”: ” 在地图中搜索位置”,

”arguments”: [”x”,”y”,”text”],

”arguments”: [”x”,”y”,”text”],

”description”: ”Tap the search bar in Google Maps at position (x, y), type the location text, and select the first search result to display the route options.”,

”description”: ” 在 Google 地图中点击位置 (x, y) 的搜索栏，输入位置文本，并选择第一个搜索结果以显示路线选项。”,

”precondition”: ”The Google Maps app is open, and the search bar is visible on the screen.”,

”precondition”: ”Google 地图应用已打开，且搜索栏在屏幕上可见。”,

”atomic_action_sequence”: [

”atomic_action_sequence”: [

{”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”}},

{”name”:” 点击”,”arguments_map”:{”x”:”x”,”y”:”y”}},

{”name”:”Type”,”arguments_map”:{”text”:”text”}},

{”name”:” 输入”,”arguments_map”:{”text”:”text”}},

{”name”:”Enter”,”arguments_map”:{}},

{”name”:” 回车”,”arguments_map”:{}},

{”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”}} < ” Redundant Tap action

{”name”:” 点击”,”arguments_map”:{”x”:”x”,”y”:”y”}} < ” 多余的点击动作

1

{

”name”: ”Switch_App_And_Search”,

”name”: ” 切换应用并搜索”,

”arguments”: [”app_name”, ”x”, ”y”, ”text”],

”arguments”: [” 应用名称”, ”x”, ”y”, ” 文本”],

”description”: ”Switch to a specified app, tap on a search bar at position (x, y), type the given text, and press Enter to perform a search.”,

”description”: ” 切换到指定应用，点击位于 (x, y) 位置的搜索栏，输入给定文本，按回车执行搜索。”,

”precondition”: ”The app to switch to is already open in the app switcher, and the search bar is visible on the screen after switching.”,

”precondition”: ” 切换的应用已在应用切换器中打开，切换后搜索栏在屏幕上可见。”,

”atomic_action_sequence”: [

”atomic_action_sequence”: [

{”name”:”Switch_App”,”arguments_map”:{}},_____Missing an additional Tap

{”name”:” 切换应用”,”arguments_map”:{}},_____ 缺少一次额外的点击

{”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”}},_____ action to get into the App

{”name”:” 点击”,”arguments_map”:{”x”:”x”,”y”:”y”}},_____ 进入应用的操作

{”name”:”Type”,”arguments_map”:{”text”:”text”}},

{”name”:” 输入”,”arguments_map”:{” 文本”:”text”}},

{”name”:”Enter”,”arguments_map”:{}}

{”name”:” 回车”,”arguments_map”:{}}

]
}

Figure 11. Example of imperfect (above) and erroneous (below) generated Shortcuts. The ”Search_Location_in_Maps” Shortcut includes an unnecessary Tap action in the operation sequence, while the ”Switch_App_And_Search” Shortcut omits a Tap action needed to first enter the desired App before performing the search.

图 11. 不完善 (上) 和错误 (下) 生成快捷指令的示例。“地图中搜索位置”(Search_Location_in_Maps) 快捷指令在操作序列中包含了不必要的点击动作，而“切换应用并搜索”(Switch_App_And_Search) 快捷指令则遗漏了进入目标应用前所需的点击动作。

Table 7. All task queries in Mobile-Eval-E.

表 7. Mobile-Eval-E 中的所有任务查询。

Scenario	Task ID	APPs	Input Query
Restaurant Recommendation	1_late_night_korean_food	Maps	Find the best-rated late-night Korean restaurant in Champaign, IL that opens beyond 9pm on Google Maps.
	1_nearest_bakery	Maps	Get directions to the nearest Bakery that has a rating higher than 4.0 on Google Maps. Stop at the screen showing the route.
	1-thai_duck	Maps, Notes	Find the best-rated Thai restaurant in Urbana, IL that serves duck cuisine on Google Maps. Review customer comments and compile a summary of positive and negative feedback in Notes.
	1_bakery_birthday_cake	Maps, Notes	Find me a Bakery that is within 10 min drive near me and does birthday cakes on Google Maps. Find the phone number and create a new note in Notes for that.
	1_chinese_share	Maps, X, Notes	Find me a popular Chinese restaurant near Chicago O'Hare airport on Google Maps. Check X for recent posts about their signature dishes and write a summary in Notes. Then get directions to that restaurant on Google Maps. Stop at the screen showing the route.
Information Researching	2_segment_anything_cited	Chrome	Find the most-cited paper that cites the paper 'Segment Anything' on Google Scholar. Stop at the screen showing the paper abstract.
	2_llm_agents_survey	Chrome, Notes	Find at least three representative survey papers on LLM agents on Google Scholar, and add their titles to the Notes.
	2_recipes_chinese	Chrome, YouTube	I have some onions, beef, and potatoes in my refrigerator. Can you find me a Chinese-style recipe that uses all three ingredients and can be prepared in under an hour? And find me a video tutorial on YouTube for that. Stop at the screen displaying the video.
	2_mcdonalds_deals	McDonald's, Maps	Can you check the McDonald's APP to see if there are any Rewards or Deals including Spicy McCrispy. If so, help me add that to Mobile Order (Do not pay yet, I will do it myself). And then check the pickup location and get directions on Google Maps. Stop at the screen showing the route.
	2-headphones_reviews	Amazon, Notes	Find three detailed user reviews of the Bose QC45 headphones from Amazon. Summarize the general sentiment in the Notes.
Online Shopping	3_oled_tv	Best Buy	Find the best deal on a 55-inch 4K OLED TV at Best Buy. Stop at the screen displaying the best deal you find.
	3_laptop_nvidia_gpu	Amazon Shopping	Find me a laptop on Amazon that is under $1000 with an Nvidia GPU and more than 8GB RAM.
	3_ninja_air_fryer	Amazon Shopping, Walmart	Compare the price of a Ninja air fryer 8 qt at Walmart and Amazon. Stop at the screen displaying the best deal you find.
	3_walmart_sale_items	Walmart, Notes	Check if any of the following items are on sale at Walmart: ribeye steak, fresh oranges, or toilet paper. If any are on sale, add a note in Notes with their prices.
	3_nintendo_switch_joy_con	Amazon Shopping, Best Buy, Walmart	I want to buy a brand-new Nintendo Switch Joy-Con. Any color is fine. Please compare the prices on Amazon, Walmart, and Best Buy. Find the cheapest option and stop at the screen where I can add it to the cart.
What's Trending	4_x_black_myth_wukong	X, Notes	Find the top posts about the game 'Black Myth Wukong' on X and summarize the key highlights in Notes.
	4_x_trending_news	X, Notes	Check the top 3 trending news on X. Read a few posts to figure out what's happening. And create a new Note to summarize your findings.
	4_watercolor_painting_tutorial	Lemon8, Notes	I want to learn how to paint watercolor. Find me some content creators to follow on Lemon8 tha has highly liked posts about watercolor painting tutorials. List their account names in Notes.
	4_movie_trending	Fandango, Notes	Check the top 5 trending movies on Fandango that are currently in theaters. Compare their ratings and create a note in Notes for the highest-rated one, including its name and showtimes.
	4_horror_movie_reviews	Fandango. Lemon8, Notes	Find me the latest horror movie currently in theaters on Fandango. Check some reviews on Lemon8 about the movie and create a note in Notes with the general sentiment.
Travel Planning	5.cheap_flights_newyork	Booking	Find the cheapest round-trip flight from Chicago to New York City in the next month on Booking. Stop at the screen showing the best deal.
	5_things_to_do_la	Tripadvisor, Notes	Suggest some interesting things to do in LA. Find the top 3 attractions on Tripadvisor. Save the list in Notes.
	5_palo_alto_tour	Tripadvisor, Notes	Plan a one-day itinerary for Palo Alto, CA using Tripadvisor. Choose the attractions and dining recommendations, but keep in mind that I don't like seafood and I love museums. Write the plan in Notes.
	5_local_food_chicago	Tripadvisor, Notes	Find a highly recommended local restaurant in Chicago on Tripadvisor. Check the reviews about must-try dishes and summarize in Notes.
	5_hotel_champaign	Booking, Maps	Help me find a hotel in Champaign, IL on Booking that is under $200 for a queen bed. Make sure that the rating is higher than 7.0. Double-check on Google Maps to see if it is close to Green Street. Show me your final choice on Booking.

场景	任务 ID	应用程序	输入查询
餐厅推荐	1_深夜韩餐	地图	在 Google 地图上查找伊利诺伊州尚佩恩市评分最高且营业时间超过晚上 9 点的深夜韩餐厅。
	1_最近的面包店	地图	获取前往 Google 地图上评分高于 4.0 的最近面包店的路线。停留在显示路线的界面。
	1_泰式鸭肉	地图，笔记	在 Google 地图上查找伊利诺伊州厄巴纳市评分最高且提供鸭肉菜肴的泰国餐厅。查看顾客评论并在笔记中汇总正面和负面反馈。
	1_面包店生日蛋糕	地图，笔记	帮我在 Google 地图上找一家距离我开车距离在 10 min 以内且做生日蛋糕的面包店。找到电话号码并在笔记中创建新条目。
	1_芝加哥奥黑尔机场附近中餐	地图，X，笔记	在 Google 地图上查找芝加哥奥黑尔机场附近受欢迎的中餐厅。查看 X 上关于其招牌菜的最新帖子，并在笔记中写总结。然后在 Google 地图上获取该餐厅的路线，停留在显示路线的界面。
信息检索	2_引用 Segment Anything 的论文	Chrome 浏览器	在 Google 学术上查找引用论文《Segment Anything》的被引用次数最多的论文。停留在显示论文摘要的界面。
	2_大型语言模型代理调查	Chrome 浏览器，笔记	在 Google 学术上查找至少三篇关于大型语言模型 (LLM) 代理的代表性综述论文，并将其标题添加到笔记中。
	2_中式食谱	Chrome 浏览器，YouTube	我冰箱里有洋葱、牛肉和土豆。帮我找一个使用这三种食材且烹饪时间不超过一小时的中式食谱，并在 YouTube 上找一个相关的视频教程。停留在显示视频的界面。
	2_麦当劳优惠	麦当劳 APP，地图	帮我查看麦当劳 APP 是否有包含辣味麦脆鸡的奖励或优惠。如果有，帮我添加到手机订单 (暂不付款，我自己操作)。然后查看取餐地点并在 Google 地图上获取路线，停留在显示路线的界面。
	2_耳机评价	亚马逊，笔记	在亚马逊上找到三条关于 Bose QC45 耳机的详细用户评价，并在笔记中总结总体评价倾向。
在线购物	3_OLED 电视	Best Buy	在 Best Buy 查找 55 英寸 4K OLED 电视的最佳优惠。停留在显示最佳优惠的界面。
	3_带 Nvidia 显卡的笔记本	亚马逊购物	帮我在亚马逊找一台价格低于 1000 美元、配备 Nvidia 显卡且内存超过 8GB 的笔记本电脑。
	3_Ninja 空气炸锅	亚马逊购物，沃尔玛	比较沃尔玛和亚马逊上 Ninja 8 夸脱空气炸锅的价格。停留在显示最佳优惠的界面。
	3_沃尔玛促销商品	沃尔玛，笔记	检查沃尔玛是否有以下商品促销: 肋眼牛排、新鲜橙子或卫生纸。如有促销，记录价格并添加到笔记中。
	3_任天堂 Switch 手柄	亚马逊购物，Best Buy，沃尔玛	我想买一套全新的任天堂 Switch 手柄，颜色不限。请比较亚马逊、沃尔玛和 Best Buy 的价格，找到最便宜的选项并停留在可以加入购物车的界面。
热门趋势	4_X 平台黑神话悟空	X 平台，笔记	查找 X 平台上关于游戏《黑神话悟空》的热门帖子，并在笔记中总结主要亮点。
	4_X 平台热门新闻	X 平台，笔记	查看 X 平台上的前三条热门新闻。阅读部分帖子了解事件详情，并新建笔记总结发现。
	4_水彩画教程	Lemon8，笔记	我想学习水彩画。帮我在 Lemon8 上找一些发布了高点赞水彩画教程内容的创作者，并将他们的账号名列入笔记。
	4_电影热映	Fandango，笔记	查看 Fandango 上当前影院上映的五部热门电影。比较评分，并在笔记中记录评分最高电影的名称和放映时间。
	4_恐怖电影评论	Fandango，Lemon8，笔记	帮我找一下 Fandango 上目前正在上映的最新恐怖电影。在 Lemon8 上查看一些关于这部电影的评论，并在笔记中写下总体评价。
旅行计划	5_纽约廉价机票	Booking	在 Booking 上查找下个月从芝加哥飞往纽约市的最便宜往返机票。停留在显示最佳优惠的页面。
	5_洛杉矶必做事项	Tripadvisor，笔记	推荐一些洛杉矶有趣的活动。在 Tripadvisor 上找出排名前三的景点。将列表保存到笔记中。
	5_帕洛阿尔托游览	Tripadvisor，笔记	使用 Tripadvisor 规划加州帕洛阿尔托的一日游行程。选择景点和餐饮推荐，但请注意我不喜欢海鲜，喜欢博物馆。将计划写入笔记。
	5_芝加哥当地美食	Tripadvisor，笔记	在 Tripadvisor 上找一家芝加哥备受推荐的本地餐厅。查看必尝菜品的评论并在笔记中总结。
	5_尚佩恩酒店	Booking，地图	帮我在 Booking 上找一家尚佩恩 (伊利诺伊州) 价格低于 200 美元、配备大号床的酒店。确保评分高于 7.0。再用谷歌地图确认是否靠近格林街。最后在 Booking 上展示你的选择。

Table 8. Atomic operations space.

表 8. 原子操作空间。

Operation	Description
Open_App(app_name)	If the current screen is Home or App screen, you can use this action to open the app named "app_name" on the visible on the current screen.
Tap (x, y)	Tap the position (x, y) in current screen.
Swipe (x_1, y_1, x_2, y_2)	Swipe from position (x_1, y_1) to position (x_2, y_2). To swipe up or down to review more content, you can adjust the y-coordinate offset based on the desired scroll distance. For example, setting $x_1 = x_2 = 0.5*$ width, $y_1 = 0.5*$ height, and $y_2 = 0.1*$ height will swipe upwards to review additional content below. To swipe left or right in the App switcher screen to choose between open apps, set the x-coordinate offset to at least $0.5 *$ width.
Type(text)	Type the "text" as an input box.
Enter()	Press the Enter key after typing (useful for searching).
Switch_App()	Show the App switcher for switching between opened apps.
Back()	Return to the previous state.
Home()	Return to home page.
Wait()	Wait for 10 seconds to give more time for a page loading.

操作	描述
打开应用 (Open_App(app_name))	如果当前屏幕是主屏幕或应用屏幕，可以使用此操作打开当前屏幕上可见的名为 "app_name" 的应用。
Tap (x, y)	点击当前屏幕上的位置 (x, y)。
Swipe (x_1, y_1, x_2, y_2)	从位置 (x_1, y_1) 滑动到位置 (x_2, y_2)。要向上或向下滑动以查看更多内容，可以根据所需滚动距离调整 y 坐标偏移。例如，设置 $x_1 = x_2 = 0.5*$ 宽度，$y_1 = 0.5*$ 高度，和 $y_2 = 0.1*$ 高度将向上滑动以查看下方的更多内容。要在应用切换器屏幕中左右滑动以选择打开的应用，x 坐标偏移应至少为 $0.5 *$ 宽度。
输入 (Type(text))	在输入框中输入 "text"。
回车 (Enter())	输入后按回车键 (用于搜索)。
切换应用 (Switch_App())	显示应用切换器以切换已打开的应用。
返回 (Back())	返回到上一个状态。
主页 (Home())	返回主页。
等待 (Wait())	等待 10 秒，为页面加载提供更多时间。

Inital Shortcuts (User Provided)

初始快捷方式 (用户提供)

”name”: ”Tap_Type_and_Enter”,

”name”: ”Tap_Type_and_Enter”,

”arguments”: [”x”,”y”,”text”],

”arguments”: [”x”,”y”,”text”],

”description”: ”Tap an input box at position (x, y), Type the \”text \”, and then perform the Enter operation. Very useful for searching and sending messages!”,

”description”: ” 点击位置 (x, y) 处的输入框，输入“text”，然后执行回车操作。非常适合搜索和发送消息！”,

”precondition”: ”There is a text input box on the screen with no previously entered content.”,

”precondition”: ” 屏幕上有一个文本输入框，且之前没有输入内容。”,

”atomic_action_sequence”: [{”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”},},{”name”:”Type”,”arguments_map”:{”text

”atomic_action_sequence”: [{”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”},},{”name”:”Type”,”arguments_map”:{”text”

{ Agent Generated Shortcuts

{ Agent Generated Shortcuts

”name”: ”Create_New_Note”,

”name”: ”Create_New_Note”,

”arguments”: [”text”],

”arguments”: [”text”],

”description”: ”Create a new note in the Notes app and type the provided text into it.”,

”description”: ” 在备忘录应用中创建新笔记，并输入提供的文本。”,

”precondition”: ”The Notes app is open, and the ’Add’ button (orange icon with a pencil) is visible on the screen.”,

”precondition”: ” 备忘录应用已打开，且屏幕上可见“添加”按钮 (橙色铅笔图标)。”,

”atomic_action_sequence”: [{”name”:”Tap”, ”arguments_map”:{”x”:”929”,”y”:”2053”}},{”name”:”Type”,”arguments_map”

”atomic_action_sequence”: [{”name”:”Tap”, ”arguments_map”:{”x”:”929”,”y”:”2053”}},{”name”:”Type”,”arguments_map”:{

”name”: ”Search Location in Maps”,

”name”: ”Search Location in Maps”,

”arguments”: [”x”,”y”,”text”],

”arguments”: [”x”,”y”,”text”],

”description”: ”Tap the search bar in Google Maps at position (x, y), type the location text, and select the first search result to display the route options.”,

”description”: ” 点击 Google 地图中位置 (x, y) 处的搜索栏，输入地点文本，并选择第一个搜索结果以显示路线选项。”,

”precondition”: ”The Google Maps app is open, and the search bar is visible on the screen.”,

”precondition”: ”Google 地图应用已打开，且搜索栏在屏幕上可见。”,

”atomic_action_sequence”: [{”name”: ”Tap”,”arguments_map”: {”x”:”x”,”y”:”y”}, {”name”:”Type”,”arguments_map”:{”text
{”rame”:”Enter”,”arguments_map”: {}), {”name”:”Tap”,”arguments_map”:{”x”:”x”,”y”:”y”}}]

”atomic_action_sequence”: [{”name”: ” 点击”,”arguments_map”: {”x”:”x”,”y”:”y”}, {”name”:” 输入”,”arguments_map”:{”text”:”text”}, {”rame”:” 回车”,”arguments_map”: {}), {”name”:” 点击”,”arguments_map”:{”x”:”x”,”y”:”y”}}]

{

”name”: ”Swipe_to_Reveal_Content”,

”name”: ” 滑动以显示内容”,

”arguments”: [”x1”,”yl”,”x2”,”y2”],

”arguments”: [”x1”,”yl”,”x2”,”y2”],

”description”: ”Swipe from position (x1, y1) to position (x2, y2) to reveal additional content below or above on the screen.”,

”description”: ” 从位置 (x1, y1) 滑动到位置 (x2, y2)，以显示屏幕上下方的额外内容。”,

”atomic_action_sequence”: [{”name”:”Swipe”,”arguments_map”:{”x1”:”x1”,”y1”:”y1”,”x2”:”x2”,”y2”} }]

”atomic_action_sequence”: [{”name”:” 滑动”,”arguments_map”:{”x1”:”x1”,”y1”:”y1”,”x2”:”x2”,”y2”} }]

”name”: ”Clear_Search_And_Type”,

”name”: ” 清除搜索并输入”,

”arguments”: [”x clear”,”y clear”,”text”],

”arguments”: [”x clear”,”y clear”,”text”],

”description”: ”Clear the current search term by tapping the ’X’ icon and then type the new search term into the search bar.”,

”description”: ” 通过点击“X”图标清除当前搜索词，然后在搜索栏中输入新的搜索词。”,

”precondition”: ”The search bar is active, and the ’X’ icon to clear the current search term is visible on the screen.”,

”precondition”: ” 搜索栏处于激活状态，且屏幕上可见用于清除当前搜索词的“X”图标。”,

”atomic action_sequence”: [{”name”:”Tap”,”arguments_map”:{”x”:”x_clear”,”y”:”y_clear”}},{”name”:”Type”,”arguments_n

”atomic action_sequence”: [{”name”:” 点击”,”arguments_map”:{”x”:”x_clear”,”y”:”y_clear”}},{”name”:” 输入”,”arguments_map”:{”text”:”text”}}]

{

”name”: ”Save_Note_As_File”,

”name”: ” 将笔记保存为文件”,

”arguments”: [”folder_x”,”folder y”,”done_x”,”done_y”,”save_x”,”save_y”],

”arguments”: [”folder_x”,”folder y”,”done_x”,”done_y”,”save_x”,”save_y”],

”description”: ”Save a note as a file in a specified folder by selecting the folder, confirming the selection, and tapping the save button.”,

”description”: ” 通过选择文件夹、确认选择并点击保存按钮，将笔记保存为指定文件夹中的文件。”,

”atomic_action_sequence”: [{”name”:”Tap”,”arguments_map”:{”x”:”folder_y”},””roledre_y”}},{”name”:”Tap”,”arguments_

”atomic_action_sequence”: [{”name”:” 点击”,”arguments_map”:{”x”:”folder_y”},””roledre_y”}},{”name”:” 点击”,”arguments_map”:{”x”:”done_y”}

{”name”:”Tap”,”arguments_map”:{”x”:”save_x”,”y”:”save_y”})]

{”name”:” 点击”,”arguments_map”:{”x”:”save_x”,”y”:”save_y”})]

”name”: ”Switch_App_And_Search”,

”name”: ” 切换应用并搜索”,

”arguments”: [”app_name”, ”x”, ”y”, ”text”],

”arguments”: [”app_name”, ”x”, ”y”, ”text”],

”description”: ”Switch to a specified app, tap on a search bar at position (x, y), type the given text, and press Enter to perform a search.”,

”description”: ” 切换到指定应用，点击位于 (x, y) 位置的搜索栏，输入给定文本，并按回车执行搜索。”

”precondition”: ”The app to switch to is already open in the app switcher, and the search bar is visible on the screen after switching.”,

”precondition”: ” 切换的应用已在应用切换器中打开，且切换后搜索栏在屏幕上可见。”

”atomic_action_sequence”: [{”name”:”Switch_App”,”arguments_map”: { } },{”name””//”anguments_map”: {”ran”” {”text”:”text”}},{”name”:”Enter”,”arguments_map”:{}}]

”atomic_action_sequence”: [{”name”:”Switch_App”,”arguments_map”: { } },{”name””//”anguments_map”: {”ran”” {”text”:”text”}},{”name”:”Enter”,”arguments_map”:{}}]

Figure 12. Full list of Shortcuts generated by Mobile-Agent-E (with GPT-4o) after self-evolution.

图 12. Mobile-Agent-E(基于 GPT-4o) 自我进化后生成的完整快捷方式列表。

** Initial Tips (User Provided) **

** 初始提示 (用户提供) **

0. Do not add any payment information. If you are asked to sign in, ignore it or sign in as a guest if possible. Close any pop-up windows when opening an app. 1. By default, no apps are opened in the background. 2. Screenshots may show partial text in text boxes from your previous input; this does not count as an error. 3. When creating new Notes, you do not need to enter a title unless the user specifically requests it.

0. 不要添加任何支付信息。如被要求登录，忽略或尽可能以访客身份登录。打开应用时关闭所有弹窗。1. 默认情况下，后台不打开任何应用。2. 截图中可能显示之前输入的文本框部分内容，这不算错误。3. 创建新笔记时，除非用户特别要求，否则无需输入标题。

** Agent Generated Tips (Scenario 1) **

** 代理生成提示 (场景 1) **

4. When searching for restaurants or businesses, ensure the query includes specific details like location, type of cuisine, and operational hours to narrow down results effectively. 5. Always verify the operational hours of businesses to ensure they meet the user's requirements, especially for late-night or time-sensitive searches. 6. When filtering search results (e.g., by rating or distance), ensure the filter criteria are applied correctly to avoid irrelevant results. 7. Double-check this selected location or business to ensure it matches the user's requirements thoroughly. 10 If an action does not return to the expected screen, use alternative navigation methods (e.g., tapping "X" or returning to the home screen) to correct the workflow. 11. When summarizing customer feedback, include both positive and negative aspects to provide a balanced overview. 12. When retrieving contact information, onsure the details (e.g., phone number or address) are accurate and match the selected business before saving them in Notes. 13. If a task involves multiple apps (e.g., Google Maps and Notes), ensure smooth transitions between apps and verify that the required information is correctly transferred. 14. If an app fails to open or respond in the app switcher, return to the home screen and reopen the app directly to avoid delays.

4. 搜索餐厅或商家时，确保查询包含具体细节，如位置、菜系类型和营业时间，以有效缩小结果范围。5. 始终核实商家的营业时间，确保符合用户需求，尤其是深夜或时间敏感的搜索。6. 筛选搜索结果 (如按评分或距离) 时，确保正确应用筛选条件，避免无关结果。7. 仔细核对所选位置或商家，确保完全符合用户要求。10. 若操作未返回预期界面，使用替代导航方法 (如点击“X”或返回主屏幕) 纠正流程。11. 汇总客户反馈时，包含正反两方面内容，提供平衡概述。12. 获取联系信息时，确保详情 (如电话号码或地址) 准确无误且与所选商家匹配，方可保存至笔记。13. 涉及多个应用 (如 Google 地图和笔记) 的任务，确保应用间顺畅切换并核实信息正确传递。14. 若应用切换器中应用无法打开或无响应，返回主屏幕直接重新打开，避免延误。

** Agent Generated Tips (Scenario 2) **

** 代理生成提示 (场景 2) **

4. When identifying the most-cited paper or similar tasks, ensure to sort the results by citation count if the option is available. This minimizes manual scanning and reduces errors. 5. If a search action fails, verify the input text and ensure the correct search bar is targeted before retrying. Adjust the tap location if necessary. 6. When recording information from search results, ensure the details are accurate and clearly formatted to avoid confusion. 7. If a task involves multiple steps across different apps, always confirm the completion of one step Retry the action with slight adjustments if necessary. 9. When selecting a video or item from a list, ensure the title matches the intended choice to avoid selecting the wrong option. 10. If a button or option does not respond to a tap, ensure it is fully visible on the screen. Use a swipe or scroll action to adjust the view if necessary before retriying. II. When switching between apps, ensure the correct app is selected from the app switcher to avoid unnecessary navigation errors. 12. Always stop at the final screen requested by the user, ensuring the task is fully completed before ending the interaction.

4. 识别被引用次数最多的论文或类似任务时，若有选项，确保按引用次数排序，减少手动筛查和错误。5. 搜索失败时，核实输入文本并确保定位正确的搜索栏后重试，必要时调整点击位置。6. 记录搜索结果信息时，确保详情准确且格式清晰，避免混淆。7. 涉及多个步骤跨应用的任务，始终确认完成每一步，必要时稍作调整后重试。9. 从列表中选择视频或项目时，确保标题与预期匹配，避免选错。10. 按钮或选项无响应时，确认其完全显示在屏幕上，必要时滑动或滚动调整视图后重试。11. 切换应用时，确保从应用切换器中选择正确应用，避免不必要的导航错误。12. 始终停留在用户请求的最终界面，确保任务完全完成后结束交互。

** Agent Generated Tips (Scenario 3) **

** 代理生成提示 (场景 3) **

4. When identifying the best deal, prioritize both price and features, and sense any discounts or promotions are clearly noted. 5. Always confirm that the displayed product matches the search criteria (e.g., size, specifications) to avoid selecting an incorrect item. 6. If the task requires stopping at a specific screen, ensure the screen is fully loaded and all relevant details are visible before stopping. 7. If a filter does not apply correctly, try adjusting it again by swiping a trapping alternative areas of the screen to reveal hidden options. 8. When using sliders for filters (e.g., price ange), swirping s often more effective than trappe to adjust the values. 9. If a filter unexpected resets or erroneous reliefs, reapply it then verify ensure that the product model and seedifications (e.g., size, features) are identical to avoid inaccurate comparisons. 12. If simpling to reveal content ensure the swipe is smooth and covers enough distance to load all relevant details on the screen. 13. If an app fails to open or navigate correctly, return to the home screen and retry the action. This of the seconds issues. 14. If a tap action does not work as expected, consider tapping alternative areas of the screen, such as associated buttons or options, to achieve the desired outcome. 15. Whe switching between apps, ensure the correct app is reopened and verify the screen before proceeding to avoid unnecessary repetition.

4. 在识别最佳优惠时，应优先考虑价格和功能，并确保任何折扣或促销信息清晰标注。5. 始终确认显示的产品符合搜索条件 (例如尺寸、规格)，以避免选择错误的商品。6. 如果任务要求停留在特定界面，确保界面完全加载且所有相关细节可见后再停止。7. 如果筛选器未正确应用，尝试通过滑动屏幕的其他区域以显示隐藏选项来重新调整。8. 使用滑块调整筛选条件 (例如价格区间) 时，滑动通常比点击更有效。9. 如果筛选器意外重置或出现错误，重新应用后核实产品型号和规格 (例如尺寸、功能) 是否一致，以避免不准确的比较。12. 若需滑动以显示内容，确保滑动动作流畅且距离足够，以加载屏幕上的所有相关细节。13. 如果应用无法正常打开或导航，返回主屏幕后重试该操作。这通常是暂时性问题。14. 如果点击操作未按预期生效，尝试点击屏幕的其他相关区域，如关联按钮或选项，以实现预期效果。15. 切换应用时，确保重新打开正确的应用并核实界面后再继续操作，以避免不必要的重复。

** Agent Generated Tips (Scenario 4) **

** 代理生成提示 (场景 4) **

4. When navigating apps, ensure that the correct ions in taped by carefully identifying its position and function to avoid misalignment or unintended actions. 5. If a search filter is applied unintentionality, clear it by tapping the ”X” coin in the search bar before proceeding with a new search. 6. Always verify the context of the search results to ensure they align with the intended query before summarizing or proceeding to the next step. 7. When recording information in Notes, ensure the formating is clear and consistent for easy readability. 8. Double-check the accuracy of the recorded information (e.g., account names, titles) before saving the note to avoid errors. 9. If reduced to an unit-ended page (e.g., ”My Orders”), navigate back to the main interact or intended section before proceeding. 10. When companing multiple items (e.g., move triangs), keep track of all relevant data to ensure accurate the correct app to avoid confusion. 12. When entering search terms, ensure the previous query is cleared completely to prevent appending incorrect text to the new query. 13. In misaligned tap opens an unintended menu (e.g., Filters), close it immediately and retry the intended action. 12. U. Sie broader search terms if specific queries fail to yield results, and retry. the search gradually based on the context. 15. If an app fails to execute a search or action, consider switching to a browser or alternative app to complete the task.

4. 在导航应用时，确保通过仔细识别位置和功能来正确点击目标，避免错位或误操作。5. 如果无意中应用了搜索筛选器，点击搜索栏中的“X”图标清除后再进行新的搜索。6. 始终核实搜索结果的上下文，确保其与预期查询一致后再进行总结或下一步操作。7. 在笔记中记录信息时，确保格式清晰一致，便于阅读。8. 保存笔记前，仔细核对记录信息的准确性 (例如账户名、标题)，避免错误。9. 如果进入了无后续操作的页面 (例如“我的订单”)，返回主界面或预期的部分后再继续。10. 比较多个项目时 (例如移动三角形)，跟踪所有相关数据以确保准确。12. 输入搜索词时，确保完全清除之前的查询，防止错误文本附加到新查询中。13. 如果误触打开了非预期菜单 (例如筛选器)，立即关闭并重试预期操作。12. 如果具体查询无结果，尝试使用更广泛的搜索词，并根据上下文逐步调整搜索。15. 如果应用无法执行搜索或操作，考虑切换到浏览器或其他应用完成任务。

** Agent Generated Tips (Scenario 5) **

** 代理生成提示 (场景 5) **

4. Always confirm that the displayed results match the search criteria (e.g., correct cities, dates, and round-trip selection) before proceeding to the next step. 5. If multile options are displayed, ensure the cheapest or most relevant option is clearly identified and selected as per the task requirements. 6. If a ”Back” button fails to function as expected, consider alternative methods to save or exit, such as using a menu or additional options (e.g., ”Save as file”). 7. When saving a note as a file, ensure the correct folder and file format are selected before confirms the save. 8. Double-check that the task is fully completed (e.g., the note is saved in the correct location) before marking its a done. 9. If serelling through content does not reveal no information, consider alternative methods to locate the required details, such as using a search or filter functions within the paper. 10. If the end of a section is reached and the required information is not found, reassess the search centerina or explore other sections of the app for relevant details. I. It when searching for selefic items (e.g., dishes, anematics). use keywords of filters to narrow down results and save time. 12. If repetitive actions (e.g., swipping) fail to yield results, pause and evaluate wither the task can be completed using different approach or if the information is unavailable. 13. When switching between approach, results in the context of the task is maintained, and verify that the information gathered in one app aligns with the requirements in the other app. 14. Always confirm the proximity or location details (e.g., using Google Maps) before finalizing a selection, especially when location is a key criterion.

4. 始终确认显示结果符合搜索条件 (例如正确的城市、日期和往返选择) 后再进行下一步。5. 如果显示多个选项，确保根据任务要求明确识别并选择最便宜或最相关的选项。6. 如果“返回”按钮未按预期工作，考虑使用菜单或其他选项 (例如“另存为文件”) 来保存或退出。7. 保存笔记为文件时，确认选择了正确的文件夹和文件格式后再确认保存。8. 在标记任务完成前，仔细检查任务是否完全完成 (例如笔记是否保存到正确位置)。9. 如果滚动内容未显示所需信息，考虑使用搜索或筛选功能等替代方法查找所需细节。10. 如果到达某部分末尾仍未找到所需信息，重新评估搜索范围或探索应用的其他部分以获取相关信息。11. 搜索特定项目 (例如菜品、电影) 时，使用关键词或筛选器缩小结果范围，节省时间。12. 如果重复操作 (例如滑动) 无效，暂停并评估是否可以用其他方法完成任务或信息是否不可用。13. 切换应用时，确保任务相关的上下文保持一致，并核实一个应用中收集的信息是否符合另一个应用的要求。14. 在最终确定选择前，始终确认距离或位置详情 (例如使用谷歌地图)，尤其当位置是关键条件时。