PERCU: Benchmarking Multimodal Agents on Personalized Computer Use Tasks under Diverse Memory Mechanisms

Anonymous ACL submission

Abstract

Large language model (LLM) agents have demonstrated remarkable potential in automating digital workflows through multimodal planning and reasoning. However, providing truly personalized assistance in computer use remains a significant challenge, as existing benchmarks predominantly treat agents as context-independent executors, operating under a homogeneity assumption that ignores the diverse user-specific habits and procedural routines. To address these challenges, we propose PERCU, a benchmark designed to evaluate the personalized capabilities of multimodal agents on computer use tasks. Besides, we introduce a comprehensive taxonomy of nine distinct memory mechanisms for personalized agents. Extensive evaluation on PERCU systematically investigates how these diverse memory mechanisms affect task accuracy. Our findings reveal significant deficiencies in their ability to serve as personalized computer assistants, providing valuable insights for future research in developing personalized multimodal agents. Our code and data will be available at https://github.com/scm62519/PERCU.

1 Introduction

The rapid advancement of Large Language Models (LLMs) has catalyzed a paradigm shift in artificial intelligence, transitioning agents from passive command-followers to autonomous entities endowed with sophisticated reasoning, planning, and reflective capabilities (Liu et al., 2023; Mialon et al., 2023). In the domain of computer use, these agents are increasingly expected to navigate complex graphical user interfaces (GUIs) to automate diverse digital workflows. However, data scarcity remains a primary bottleneck. While users demand highly tailored assistance, acquiring large-scale, high-quality labeled data for every unique individual to train a personalized assistant is practically infeasible. This tension necessitates a move toward few-shot personalization (Kim and Yang, 2025), where an agent must rapidly adapt to a user's specific preferences and behavioral patterns from minimal interaction history. Personalization (Liu et al., 2025; Nandakishor and Anjali, 2025; Shenfeld et al., 2025) represents the critical evolution of an agent from a generic automation tool into a true digital companion. Such an agent functions as a partner that understands underlying intent and idiosyncratic habits rather than a cold executor of literal, context-free commands (Wu et al., 2025).

Despite the burgeoning interest in computer use agents, current evaluative frameworks, such as OSWorld (Xie et al., 2024b) and WebArena (Zhou et al., 2023), largely overlook the subjective and repetitive nature of human-computer interaction. These benchmarks typically operate under a homogeneity assumption, where success is measured against a single, universal execution path. Such a paradigm ignores the fact that a user's historical preferences and long-term routines are decisive factors in task execution. For instance, in a personalized educational or research setting, an agent should not only execute a download but also recognize that a specific user consistently categorizes academic PDFs into a "paper" folder and subsequently initiates a note-taking routine. As illustrated in Figure 1, the failure of existing models to integrate such personalized knowledge results in a significant gap between laboratory performance and real-world utility, highlighting the urgent need for a benchmark that accounts for personalized computer use tasks.

Existing multimodal agents lack a standardized benchmark to test their ability to internalize digital habits, such as specific file organization patterns or repetitive software routines, which are essential for transforming an automation tool into a truly personalized assistant. To bridge this gap, we propose PERCU (**PER**sonalization in **C**omputer **U**se tasks), the first benchmark specifically designed to assess

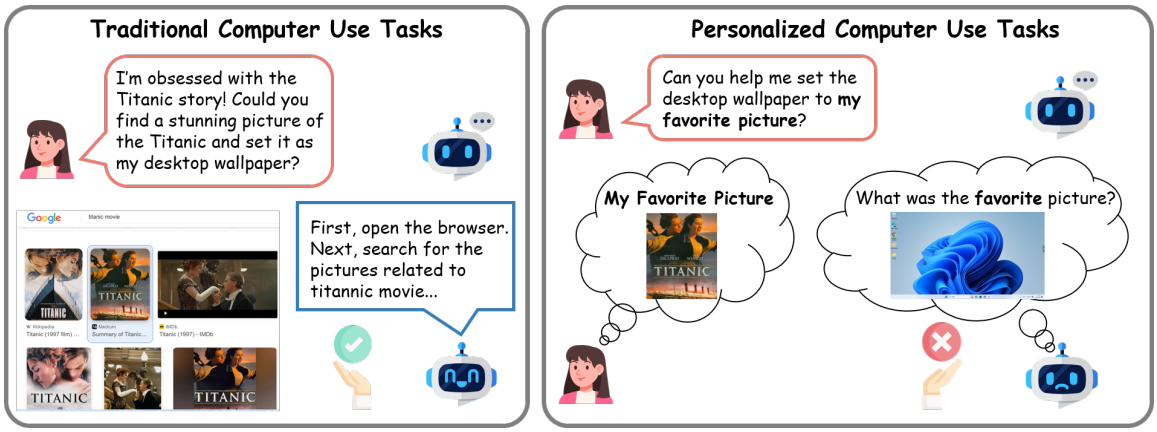

Figure 1: Comparison between traditional computer use tasks and personalized computer use tasks. While conventional agents focus on strictly following literal instructions based on universal defaults, personalized assistance agents must leverage user-specific knowledge grounded in past interactions. This disparity highlights the critical necessity of transitioning from rigid instruction-following toward context-aware, long-term personalized assistants that can resolve ambiguity through behavioral priors.

the personalized capabilities of multimodal agents on computer use tasks. PERCU fills this gap by formalizing and evaluating personalized computer use tasks via a dual-instruction framework. Recognizing that the success of this paradigm hinges on how historical context is managed, we fundamentally introduce a comprehensive framework of nine distinct memory mechanisms for personalized agents. To systematically evaluate cognitive demands, we categorize these memory mechanisms into three dimensions: **Information Distillation and Filtering**, **Relational Reasoning and Structuring**, and **Execution Constraints and Retrieval**. Our main contributions are summarized as follows:

- We formalize the definition of personalization within the computer use paradigm, categorizing personalized tasks into two distinct types: preference tasks, which capture static user inclinations, and routine tasks, which represent dynamic, multi-step procedural workflows.
- We propose the PERCU benchmark, a comprehensive evaluation suite that requires agents to leverage memory to resolve personalized instructions in realistic desktop environments.
- We introduce a taxonomy of nine memory mechanisms tailored for personalized agents. Systematically classified across three cognitive dimensions, this framework is specifically developed to enable accurate personalized memory retrieval and dynamic long-term memory management in realistic computer use environments.
- Through extensive experimentation and in-depth analysis of multimodal agents equipped with these diverse memory mechanisms, we identify critical bottlenecks in current models, demonstrating how targeted memory mechanisms can drastically affect task accuracy.

2 Related Work

Benchmarks for Agents. The emergence of LLMs as autonomous agent cores (Hong et al., 2024; Niu et al., 2024; Cai et al., 2025; Qin et al., 2025a) has necessitated the development of rigorous evaluative frameworks to measure their reasoning and decision-making capabilities in interactive environments. Early efforts focused on narrow domains such as tool utilization (Qin et al., 2023) or coding tasks. More recently, multi-dimensional benchmarks like AgentBench (Liu et al., 2023) have been introduced to provide a comprehensive assessment across diverse settings, including operating systems, databases, and knowledge graphs. GAIA (Mialon et al., 2023) shifted the focus toward tasks that are conceptually simple for humans but challenging for AI, emphasizing tool-use proficiency and common-sense reasoning. Furthermore, TravelPlanner (Xie et al., 2024a) highlighted the limitations of current agents in handling long-horizon planning with complex constraints. Moving beyond the browser, OSWorld (Xie et al., 2024b) and WindowsAgentArena (Bonatti et al., 2024) introduced unified frameworks for controlling entire operating systems, focusing on cross-application workflows. WebArena (Zhou et al.,

2023) established a milestone by providing a realistic, self-hosted web environment for evaluating agents on end-to-end tasks. This was followed by Mind2Web (Deng et al., 2023), which introduced a large-scale dataset for general-domain web navigation. While these benchmarks have significantly advanced the field, they primarily evaluate agents as generalist executors and focus on functional correctness within a zero-shot context, lacking a mechanism to evaluate how agents adapt to the evolving routines or preferences of a specific user and ignoring the idiosyncratic habits that define real-world human-agent collaboration. Here, we provide a comprehensive comparison between PERCU and several existing representative benchmarks for agents, as illustrated in Table 1.

Personalization in LLMs. Personalization in LLMs (Zhang et al., 2024; Woźniak et al., 2024; Tan et al., 2024; Chen et al., 2024) has traditionally focused on linguistic style mimicry or topical preference alignment in conversational settings. The LaMP benchmark (Salemi et al., 2024) introduced a systematic way to evaluate personalized text classification and generation by providing models with user-specific history. In the realm of safety, PENGUIN (Wu et al., 2025) explored personalized safety alignment, demonstrating that an agent's risk assessment should vary based on the user's unique background. Closer to our work, MEMENTO (Kwon et al., 2025) investigated how embodied agents utilize episodic memory to solve personalized object rearrangement tasks. Despite these advancements, personalization in the computer use domain remains largely underexplored.

3 PERCU

3.1 Definition of Personalization in Computer Use Tasks

Personalization, in the broader context of artificial intelligence, refers to the tailoring of information, services, or system behaviors to accommodate the specific requirements and preferences of individual users or distinct user groups (Pazzani and Billsus, 2007). While the fundamental objective remains the enhancement of system relevance and user satisfaction, the manifestation of personalization varies significantly across domains. In conversational systems, personalization is primarily characterized by the alignment of agent responses with a user's persona, linguistic style, and historical interaction context (Li et al., 2016). Besides, in recommendation systems, the emphasis lies in the dynamic generation of content suggestions, such as advertisements or products, derived from latent behavioral patterns (Resnick and Varian, 1997). Furthermore, personalization in human-computer interaction focuses on the system's adaptation to usage habits, such as optimized interface layouts or specialized shortcut configurations, whereas in embodied AI, it necessitates the continuous adjustment of physical action strategies based on long-term human-robot interactions (Kwon et al., 2025).

In the specific domain of computer use, personalization transcends simple content filtering and enters the realm of complex workflow adaptation. We formalize personalization within this paradigm through five core dimensions that collectively influence how an agent interacts with a desktop environment.

The first dimension is **User Attributes**, which encompass static profiles such as occupation, age, cultural background, cognitive style, and personality trait. These attributes dictate the fundamental behavioral logic of the agent. Moreover, agents must account for physical or cognitive constraints, such as adjusting visual interfaces for users with color vision deficiencies or prioritizing voice-based commands for users with motor impairments.

The second and third dimensions focus on **Interests** and **Behavioral Habits**, respectively. Interests determine topical preferences and the content-level priority of an agent, such as prioritizing academic tools over social software or opting for deep technical summaries over brief abstracts during literature review tasks. These preferences often extend to functional choices, such as a user's predilection for command line interfaces over GUIs. Complementing these interests are behavioral habits, which manifest as repetitive usage patterns and interaction frequencies. These include the frequent use of specific document templates, preferred window management configurations, or idiosyncratic keyboard shortcuts.

The fourth and fifth dimensions, **Task Execution Flow** and **Decision-making Style**, represent the procedural and strategic layers of personalization. Task execution flow describes the specific sequence of actions a user employs to achieve a goal, such as the distinct routine of searching, downloading, highlighting, and summarizing a research paper. Understanding these idiosyncratic routines allows the agent to move beyond reactive command following to proactive workflow automation. This is

Benchmark	Environment	Tasks	Personalization	Instruction Type	Paradigm
Mind2Web (Deng et al., 2023)	Web (Cross-site)	2350	None	Explicit	Single-turn
WebArena (Zhou et al., 2023)	Web (Self-host)	812	None	Explicit	End-to-end
VisualWebArena (Koh et al., 2024)	Web (Visual)	910	None	Explicit	End-to-end
OSWorld (Xie et al., 2024b)	Cross-OS	369	None	Explicit	End-to-end
Windows Agent Arena (Bonatti et al., 2024)	Windows OS	153	None	Explicit	End-to-end
PERCU (Ours)	**Windows Desktop**	**312**	**Preference&Routine**	**Ambiguous**	**Dual-instruction**

Table 1: Comparison of PERCU with several existing agent evaluation benchmarks. Unlike prior environments that rely on explicit instructions and lack user-specific contexts, PERCU introduces a dual-instruction paradigm. This design features ambiguous requests paired with memory mechanisms, enabling the comprehensive evaluation of multimodal agents across distinct preference and routine tasks.

further modulated by the user's decision-making style, which characterizes their tolerance for risk and exploration. Collectively, these five factors form the foundational pillars of personalized computer use, necessitating a benchmark that evaluates an agent's ability to synthesize these multi-faceted signals from the memory.

3.2 Composition of PERCU

The construction of PERCU is grounded in the high-quality trajectory data provided by the PC-Agent-E dataset (He et al., 2025), which serves as the foundational substrate for our evaluation suite. Building upon this data foundation, we introduce a dual-instruction paradigm inspired by the two-stage evaluation framework for personalized embodied assistants (Kwon et al., 2025). For every task in PERCU, we manually curate a pair of directives which include the **First Instruction** and the **Second Instruction**. The First Instruction represents the initial interaction where the user explicitly describes their task while embedding one of the five personalization dimensions discussed in Section 3.1. These instructions are semantically transparent and detailed. As illustrated in Figure 1, a user might state, "I'm obsessed with the Titanic story! Could you find a stunning picture of the Titanic and set it as my desktop wallpaper?". During this stage, the agent executes the task and ingests the resulting trajectory into its memory bank, thereby internalizing the user's specific preferences or habits as contextual knowledge.

The Second Instruction simulates subsequent interactions where user prompts become increasingly underspecified or implicit. Unlike the initial setup, these "personalized instructions" are often vague, such as "Can you help me set the desktop wallpaper to my favorite picture?". The fundamental challenge posed by PERCU is whether the agent can successfully bridge the gap between this underspecified query and the explicit knowledge stored in its memory. For example, inferring that favorite picture refers to the "Titanic" image from the previous interaction. This setup reflects realistic daily usage, where users expect an intelligent assistant to leverage the memory to resolve ambiguity without requiring repetitive, exhaustive explanations.

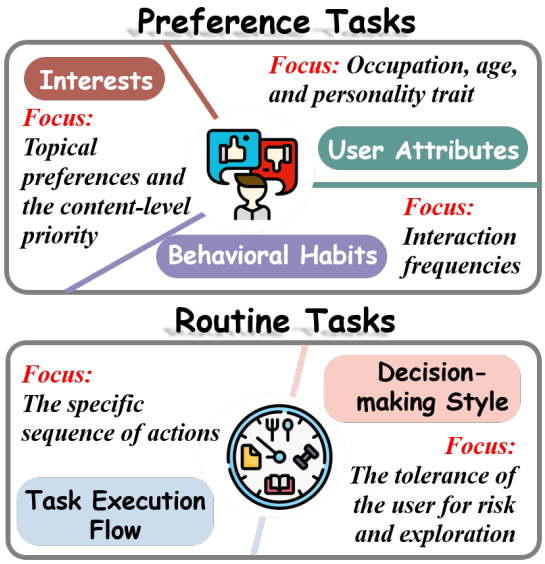

Figure 2: The illustration of two distinct types of tasks in PERCU.

To further systematize the evaluation, PERCU categorizes tasks into two distinct types which include **preference tasks** and **routine tasks**. As shown in Figure 2, preference tasks are designed to capture a user's static inclinations toward their digital environment, typically involving one-time configurations or long-term default settings. The success of an agent in these tasks hinges on its ability to ground abstract personalized entities in specific configuration instances. In contrast, routine tasks assess the agent's mastery of dynamic, multi-step procedural workflows. More details of evaluation process are provided in Appendix B.

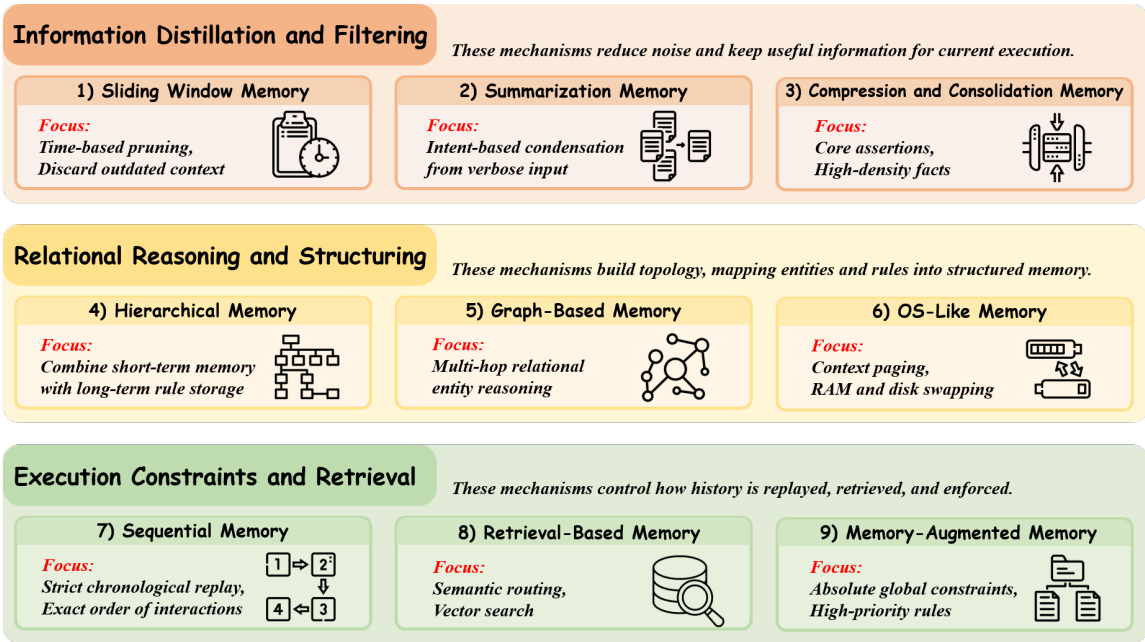

Figure 3: Taxonomy of the memory mechanisms for personalized agents, comprising three core dimensions based on their functional roles: Information Distillation and Filtering, which aims to eliminate contextual noise and extract key execution signals; Relational Reasoning and Structuring, which focuses on capturing the logical dependencies and structural integrity of tasks; Execution Constraints and Retrieval, which is designed to address precise knowledge extraction from large-scale databases and ensure strict adherence to critical user boundaries.

4 Memory Mechanisms for Personalized Agents

Drawing inspiration from the foundational architecture introduced in the open-source repository by Khan (2025), we introduce a comprehensive taxonomy comprising nine distinct memory mechanisms to systematically investigate how different memory mechanisms influence the performance of multimodal agents in personalized computer use tasks. We modified the original text-only implementations to support our dual-instruction multimodal evaluation paradigm. We categorize these nine memory mechanisms into three macroscopic dimensions based on their functional roles in processing personalized knowledge: **Information Distillation and Filtering**, **Relational Reasoning and Structuring**, and **Execution Constraints and Retrieval**, as shown in Figure 3. This taxonomy allows for a granular evaluation of agent capabilities across varying cognitive demands.

4.1 Information Distillation and Filtering

This dimension encompasses memory mechanisms designed to subtract noise from the context, ensuring that the agent focuses solely on the most relevant execution signals. Graphical interfaces often elicit trial-and-error behaviors or verbose user inputs containing emotional expressions and changing preferences. To address this, **Sliding Window Memory** operates by strictly discarding outdated conversational turns or UI states, thereby preventing the agent from being misled by early, contradictory instructions and ensuring alignment with the user's latest intent. Alternatively, **Summarization Memory** tackles verbosity by leveraging large language models to semantically condense lengthy operational histories into a clean, core workflow, omitting the user's intermediate hesitations or corrected mistakes. Taking this abstraction further, **Compression and Consolidation Memory** acts as a rigorous data distillation pipeline. Instead of preserving natural language fluency, it ruthlessly strips away all colloquialisms to extract high-density factual assertions or key-value pairs, such as specific UI parameter configurations. This extreme reduction minimizes the token burden and mitigates the risk of hallucination during long-horizon GUI navigation tasks.

4.2 Relational Reasoning and Structuring

The second dimension focuses on capturing the structural integrity and logical dependencies of personalized tasks. Many routine computer operations demand strict adherence to procedural sequences

or involve complex relationships between applications, files, and user identities. **Hierarchical Memory** bridges the gap between transient operations and persistent habits by maintaining a fast-updating working memory alongside a stable long-term vector repository. This dual-layered architecture allows the agent to dynamically combine overarching personalization rules with the immediate context of the current graphical interface. For tasks requiring multi-hop reasoning across different software ecosystems, **Graph-Based Memory** constructs explicit semantic triples from the user's historical habits, such as binding specific programming languages to particular integrated development environments. By mapping these entity relations into a directed knowledge graph, the agent can logically deduce the correct operational path even when faced with novel task instances. Finally, **Operating System-Like (OS-Like) Memory Management** introduces a sophisticated paging mechanism for agents handling multiple distinct workflows or user personas. By simulating random access memory and disk storage, this mechanism swaps entire contextual workspaces in and out based on keyword triggers, facilitating seamless transitions between disparate environments.

4.3 Execution Constraints and Retrieval

The final dimension addresses the challenges of precise knowledge extraction and strict adherence to critical user boundaries within vast personal databases. As users accumulate extensive libraries of routines across various applications, standard context windows become inadequate. **Sequential Memory** represents the most fundamental approach in this category, maintaining an unadulterated, chronological trajectory of all past operations to facilitate tasks where the exact order of execution is paramount and cannot be summarized away. Furthermore, **Retrieval-Based Memory** addresses this by embedding historical habits into a high-dimensional vector space, enabling the agent to perform semantic searches based on the current ambiguous instruction and fetch only the most pertinent past workflows without overwhelming the context limit. In contrast, **Memory-Augmented Memory** is specifically tailored for absolute, non-negotiable rules, such as privacy restrictions or accessibility requirements like high-contrast modes. It isolates these critical directives as permanent factual tokens that persistently override default system behaviors, ensuring safe and compliant execution across all tasks.

5 Experiments on PERCU

To systematically evaluate the performance of multimodal agents in personalized desktop environments, we conduct extensive experiments on the PERCU benchmark. Our evaluation focuses on assessing whether current agents can effectively leverage memory to internalize user habits and execute complex routines. We evaluate four multimodal models that represent different architectural approaches to GUI automation, providing a comprehensive overview of the current landscape of personalized computer use.

5.1 Experiments Setup

Settings. All experiments are conducted in a standardized vision-only setting, where agents observe the environment exclusively through screen captures without access to structured metadata such as document object models or accessibility trees. To ensure consistency and avoid coordinate misalignment across different models, the screen resolution for all evaluation tasks is fixed at 1280×720 pixels. This setup requires agents to possess robust visual grounding and semantic reasoning capabilities to map high-level personalized instructions to precise pixel-space coordinates.

Datasets. The dataset statistics of preference tasks and routine tasks in PERCU are shown in Table 2.

Tasks	Number of Tasks	AL of FI	AL of SI
Preference tasks	121	117.36	104.22
Routine tasks	191	97.30	88.07
All tasks	312	105.08	94.34

Table 2: The dataset statistics of preference tasks and routine tasks in PERCU. "AL" denotes the average length. "FI" denotes the First Instruction. "SI" denotes the Second Instruction.

Evaluation Metrics. To quantify agent performance, we employ two primary evaluation metrics which include **Micro Average Accuracy** and **Macro Average Accuracy**.

Macro average accuracy is computed by first calculating the success rate within each task category (i.e., preference Tasks and routine tasks) and then taking the unweighted average of these task-level scores. This ensures that the evaluation is not biased toward tasks with a disproportionately large

number of steps and accurately reflects the agent's balanced proficiency across different personalization dimensions. The macro average accuracy is defined as:

$$\text{Macro Average Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{C_i}{S_i}\right), \quad (1)$$

where N denotes the total number of tasks, i denotes the index of the i-th task, C_i denotes the number of correctly predicted steps in the i-th task, S_i denotes the total number of steps in the i-th task, and $\frac{C_i}{S_i}$ denotes the accuracy rate of the i-th task.

Micro average accuracy, on the other hand, is calculated at the step level, representing the proportion of correctly executed actions over the total number of evaluation steps across all tasks. This metric provides an overall measure of the agent's operational reliability and stability. The micro average accuracy is defined as:

$$\text{Micro Average Accuracy} = \frac{\sum_{i=1}^{N} C_i}{\sum_{i=1}^{N} S_i}. \quad (2)$$

5.2 Baselines

We benchmark four representative multimodal GUI agents to establish a competitive performance baseline. OS-Atlas-Pro-7B (Wu et al., 2024) is evaluated as a foundation action model designed for generalist GUI interactions, leveraging large-scale cross-platform pre-training to achieve superior visual perception. GUI-Owl-7B, the core of the Mobile-Agent-v3 framework (Ye et al., 2025), focuses on fundamental GUI automation through a vision-centric architecture optimized for high-fidelity screen understanding. We also include GUI-R1 (Luo et al., 2025), a generalist R1-style model that utilizes long-horizon reasoning and reinforcement learning to handle complex, multi-step interface operations. Finally, we evaluate UI-TARS-1.5-7B (Qin et al., 2025b), a native agent model designed for automated GUI interaction that emphasizes robust instruction following and real-time adaptation to diverse software environments. These baselines span a wide range of parameter scales and training paradigms, allowing for a multifaceted analysis of memory-driven personalization.

5.3 Main Results

The overall performance of the multimodal agents and humans on the PERCU benchmark is presented in Table 3 and 4, respectively. For the sake of brevity, our detailed analysis focuses primarily on the preference tasks, as they best exemplify the challenges of personalization. The details of experimental results for routine tasks are provided in Appendix C.

Among the evaluated models, GUI-Owl-7B (Ye et al., 2025) and GUI-R1-7B (Luo et al., 2025) demonstrate the most competitive capabilities across various memory settings. Notably, GUI-Owl-7B (Ye et al., 2025) achieves the highest peak performance of 35.99% macro accuracy and 35.36% micro accuracy. In contrast, OS-Atlas-Pro-7B (Wu et al., 2024) struggles significantly with personalized context, consistently yielding accuracies below 8%. However, even the most capable 7B models exhibit a substantial performance gap when compared to human performance, which reaches 91.32% macro average accuracy. This notable disparity, where human accuracy is nearly triple that of the leading agent configurations, underscores the significant challenge posed by PERCU and validates its hardness in capturing the nuances of personalized computer use.

6 Discussion

The experimental results in Table 3 and 4 provide a sobering assessment of the current state of personalized computer use agents. The results in Table 3 reveal compelling insights into how different memory mechanisms influence an agent's ability to handle preference tasks.

Information Distillation and Filtering Mechanisms such as Compression and Consolidation Memory provide the most substantial performance boost for smaller models. For instance, GUI-R1-7B (Luo et al., 2025) achieves its highest macro accuracy (29.62%) under this setting, proving that extracting high-density core assertions is crucial when an agent's native reasoning capacity or context window is intrinsically limited.

Relational Reasoning and Structuring Architectures like OS-Like Memory Management also show remarkable effectiveness (e.g., 34.63% macro accuracy for GUI-Owl-7B (Ye et al., 2025)). This indicates that structuring historical context akin to RAM and disk swapping helps the agent dynamically manage and retrieve complex, multi-layered user rules.

Execution Constraints and Retrieval Mechanisms in this dimension, particularly Sequential

Memory Mechanisms	GUI-R1-3B		GUI-R1-7B		GUI-Owl-7B		UI-TARS-1.5-7B		OS-Atlas-Pro-7B	
	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro
Information Distillation and Filtering										
Sliding Window Memory	16.68	15.78	32.41	31.58	33.26	32.13	28.62	27.98	5.47	4.52
Summarization Memory	18.65	18.01	29.69	28.53	32.13	31.39	25.64	24.75	3.73	3.32
Compression and Consolidation Memory	22.69	22.16	29.62	28.81	24.98	24.28	25.17	24.01	5.16	4.43
Relational Reasoning and Structuring										
Hierarchical Memory	16.89	15.69	29.36	28.99	31.99	31.21	24.59	23.73	6.87	5.82
Graph-Based Memory	15.91	15.33	25.19	24.09	21.29	20.97	18.96	17.82	2.11	1.75
OS-Like Memory Management	16.94	15.97	30.66	29.92	34.63	33.06	25.29	24.19	4.93	4.43
Execution Constraints and Retrieval										
Sequential Memory	21.49	20.13	31.45	31.21	35.99	35.36	26.74	25.39	7.48	6.46
Retrieval-Based Memory	15.58	15.33	30.59	28.99	21.87	21.42	25.64	24.93	4.25	3.32
Memory-Augmented Memory	17.81	16.99	30.81	29.55	32.92	32.23	28.07	27.15	1.51	1.21

Table 3: Main results of preference tasks across different memory architectures on PERCU. "Macro" and "Micro" denote Macro Average Accuracy (%) and Micro Average Accuracy (%), respectively. Higher values indicate better performance.

	Preference Tasks		Routine Tasks		All Tasks	
	Macro	Micro	Macro	Micro	Macro	Micro
Humans	91.08	90.59	91.57	90.94	91.32	90.47

Table 4: Human performance on PERCU. "Macro" and "Micro" denote Macro Average Accuracy (%) and Micro Average Accuracy (%), respectively. Higher values indicate better performance.

Memory, yield the absolute peak performances for the leading models (e.g., 35.99% for GUI-Owl-7B (Ye et al., 2025) and 31.45% for GUI-R1-7B (Luo et al., 2025)). This suggests that for preference tasks, strict chronological replay and maintaining the exact order of historical interactions are highly effective strategies for ensuring alignment with user habits.

In summary, while advanced memory mechanisms significantly elevate agent performance, the degradation across complex personalized tasks remains a primary obstacle to achieving human-level personalized assistance. More discussions about boundaries of personalized capabilities and future development of personalization are provided in Appendix E and Appendix E, respectively.

7 Conclusion

In this paper, we introduce PERCU, a benchmark designed to evaluate the personalized capabilities of multimodal agents on computer use tasks through a dual-instruction paradigm. To enable accurate personalized memory retrieval and dynamic long-term memory management in realistic computer use environments, we formalize a taxonomy of nine distinct memory mechanisms, categorizing them into Information Distillation and Filtering, Relational Reasoning and Structuring, and Execution Constraints and Retrieval. Our extensive evaluation of several multimodal agents reveals a pervasive performance deficiency in personalized computer use tasks, particularly when contrasted with the robust proficiency of human users. Moreover, the results demonstrate the effectiveness of memory mechanisms is highly task-dependent and model-specific. Ultimately, this work not only provides a rigorously challenging testbed but also establishes vital empirical guidelines for designing the next generation of memory-based and human-centered AI systems.

Limitations

Our work has the following limitations. (1) Dependency on context window length and scalability bottlenecks persist as a structural constraint. (2) The current protocol relies on step-wise independent evaluation rather than fully autonomous, long-horizon memory management.

Future iterations of PERCU will bridge these gaps by exploring the integration of fully autonomous, long-horizon retrieval-augmented generation frameworks and dynamic memory-forgetting mechanisms to adapt to continuously evolving user habits over extended operational timelines.

Ethics Statement

Ethical considerations are paramount in the development of PERCU, as evaluating personalized agents for computer use necessitates the handling of digital interaction data that could potentially contain sensitive information. To mitigate privacy risks, our benchmark is constructed upon the PC-Agent-E dataset (He et al., 2025), which was collected following strict local-only recording protocols. During the curation of PERCU, we conducted an exhaustive manual review of all tasks to ensure that any potential personally identifiable information, such as usernames, email addresses, or private file contents in the screenshots, was thoroughly redacted or replaced with synthetic placeholders. We emphasize that PERCU is intended solely for academic research to improve the helpfulness and safety of digital assistants, and we strongly discourage its use in developing systems that bypass user consent or security protocols.

Finally, the human baseline reported in our experiments was established through a controlled study involving 15 participants with backgrounds in computer science. All participants were informed of the study's objectives and provided written consent for their anonymized performance data to be used for research purposes. They were compensated with a fair hourly wage consistent with local research assistant standards. We have also considered the environmental impact of evaluating large-scale multimodal models and have optimized our evaluation pipeline to minimize redundant computations, thereby reducing the carbon footprint associated with GPU utilization.

References

Rogerio Bonatti, Dan Zhao, Francesco Bonacci, Dillon Dupont, Sara Abdali, Yinheng Li, Yadong Lu, Justin Wagle, Kazuhito Koishida, Arthur Bucker, and 1 others. 2024. Windows agent arena: Evaluating multi-modal os agents at scale. *arXiv preprint arXiv:2409.08264*.

Hongru Cai, Yongqi Li, Wenjie Wang, Fengbin Zhu, Xiaoyu Shen, Wenjie Li, and Tat-Seng Chua. 2025. Large language models empowered personalized web agents. In *Proceedings of the ACM on Web Conference 2025*, pages 198–215.

Yi-Pei Chen, Noriki Nishida, Hideki Nakayama, and Yuji Matsumoto. 2024. Recent trends in personalized dialogue generation: A review of datasets, methodologies, and evaluations. *arXiv preprint arXiv:2405.17974*.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. 2023. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114.

Ming Ding, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Cogltx: Applying BERT to long texts. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Peng Gao, Jiasen Lu, Hongsheng Li, Roozbeh Mottaghi, and Aniruddha Kembhavi. 2021. Container: Context aggregation network. *ArXiv preprint*, abs/2106.01401.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Yanheng He, Jiahe Jin, and Pengfei Liu. 2025. Efficient agent training for computer use. *arXiv preprint arXiv:2505.13909*.

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, and 1 others. 2024. Cogagent: A visual language model for gui agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14281–14290.

Maor Ivgi, Uri Shaham, and Jonathan Berant. 2023. Efficient long-text understanding with short-text models. *Transactions of the Association for Computational Linguistics*, 11:284–299.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.

Fareed Khan. 2025. Optimizing memory of ai agents. Accessed: 2026-05-24.

Jaehyung Kim and Yiming Yang. 2025. Few-shot personalization of llms with mis-aligned responses. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11943–11974.

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

Taeyoon Kwon, Dongwook Choi, Sunghwan Kim, Hyojun Kim, Seungjun Moon, Beong-woo Kwak, Kuan-Hao Huang, and Jinyoung Yeo. 2025. Embodied agents meet personalization: Exploring memory utilization for personalized assistance. *arXiv e-prints*, pages arXiv–2505.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and William B Dolan. 2016. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003.

Jiahong Liu, Zexuan Qiu, Zhongyang Li, Quanyu Dai, Wenhao Yu, Jieming Zhu, Minda Hu, Menglin Yang, Tat-Seng Chua, and Irwin King. 2025. A survey of personalized large language models: Progress and future directions. *arXiv preprint arXiv:2502.11528*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, and 1 others. 2023. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.

Run Luo, Lu Wang, Wanwei He, Longze Chen, Jiaming Li, and Xiaobo Xia. 2025. Gui-r1: A generalist r1-style vision-language action model for gui agents. *arXiv preprint arXiv:2504.10458*.

Potsawee Manakul and Mark Gales. 2021. Long-span summarization via local attention and content selection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6026–6041, Online. Association for Computational Linguistics.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*.

M Nandakishor and M Anjali. 2025. Continuous learning conversational ai: A personalized agent framework via a2c reinforcement learning. *arXiv preprint arXiv:2502.12876*.

Runliang Niu, Jindong Li, Shiqi Wang, Yali Fu, Xiyu Hu, Xueyuan Leng, He Kong, Yi Chang, and Qi Wang. 2024. Screenagent: A vision language model-driven computer control agent. *arXiv preprint arXiv:2402.07945*.

Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. 2023. Memgpt: Towards llms as operating systems.

Michael J Pazzani and Daniel Billsus. 2007. Content-based recommendation systems. In *The adaptive web: methods and strategies of web personalization*, pages 325–341. Springer.

Tianrui Qin, Qianben Chen, Sinuo Wang, He Xing, King Zhu, He Zhu, Dingfeng Shi, Xinxin Liu, Ge Zhang, Jiaheng Liu, and 1 others. 2025a. Flash-searcher: Fast and effective web agents via dag-based parallel execution. *arXiv preprint arXiv:2509.25301*.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, and 1 others. 2025b. Ui-tars: Pioneering automated gui interaction with native agents. *arXiv preprint arXiv:2501.12326*.

Nir Ratner, Yoav Levine, Yonatan Belinkov, Ori Ram, Inbal Magar, Omri Abend, Ehud Karpas, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. Parallel context windows for large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6383–6402, Toronto, Canada. Association for Computational Linguistics.

Paul Resnick and Hal R Varian. 1997. Recommender systems. *Communications of the ACM*, 40(3):56–58.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2024. Lamp: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7370–7392.

Alireza Salemi and Hamed Zamani. 2024. Evaluating retrieval quality in retrieval-augmented generation.

In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2395–2400.

Idan Shenfeld, Felix Faltings, Pulkit Agrawal, and Aldo Pacchiano. 2025. Language model personalization via reward factorization. *arXiv preprint arXiv:2503.06358*.

Zhaoxuan Tan, Zheyuan Liu, and Meng Jiang. 2024. Personalized pieces: Efficient personalized large language models through collaborative efforts. *arXiv preprint arXiv:2406.10471*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Stanisław Woźniak, Bartłomiej Koptyra, Arkadiusz Janz, Przemysław Kazienko, and Jan Kocoń. 2024. Personalized large language models. In *2024 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 511–520. IEEE.

Yuchen Wu, Edward Sun, Kaijie Zhu, Jianxun Lian, Jose Hernandez-Orallo, Aylin Caliskan, and Jindong Wang. 2025. Personalized safety in llms: A benchmark and a planning-based agent approach. *arXiv preprint arXiv:2505.18882*.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and 1 others. 2024. Os-atlas: A foundation action model for generalist gui agents. *arXiv preprint arXiv:2410.23218*.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. 2024a. Travelplanner: A benchmark for real-world planning with language agents. *arXiv preprint arXiv:2402.01622*.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, and 1 others. 2024b. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094.

Jiabo Ye, Xi Zhang, Haiyang Xu, Haowei Liu, Junyang Wang, Zhaoqing Zhu, Ziwei Zheng, Feiyu Gao, Junjie Cao, Zhengxi Lu, and 1 others. 2025. Mobile-agent-v3: Fundamental agents for gui automation. *arXiv preprint arXiv:2508.15144*.

Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, and 1 others. 2024. Personalization of large language models: A survey. *arXiv preprint arXiv:2411.00027*.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, and 1 others. 2023. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*.

Appendix Overview

Within this supplementary material, we elaborate on the following aspects:

- Appendix A: Details of Prompts
- Appendix B: Evaluation Process of PERCU
- Appendix C: Main Results for Routine Tasks on PERCU
- Appendix D: Boundaries of Personalized Capabilities
- Appendix E: Future Development of Personalization

A Details of Prompts

System prompt for PERCU is shown in Figure 4.

B Evaluation Process of PERCU

The evaluation protocol of PERCU is designed to systematically quantify the execution accuracy of multimodal LLM agents across personalized computer use trajectories. We implement an automated evaluation pipeline that iterates through the JSONL-formatted task files, traversing each logical step to assess the agent's decision-making precision. For each test instance, the agent is initialized with a meticulously crafted system prompt that defines its role as a multimodal GUI operator. This prompt establishes the behavioral constraints for interacting with graphical interfaces and mandates a chain-of-thought reasoning structure (Wei et al., 2022), requiring the model to articulate its cognitive process before generating a structured action command. This decoupling of reasoning and execution is pivotal for enhancing the logical consistency of agents in complex desktop environments. System prompt for PERCU is shown in Figure 4 of Appendix A.

A critical challenge in evaluating personalized agents is the injection of the necessary personalized knowledge to resolve the underspecified second instructions. Following the methodology of PC-Agent-E (He et al., 2025), we utilize the reconstructed thought processes that are originally generated during the thought completion stage as a form of short-term memory. As illustrated in Figure 5, the final input is constructed by concatenating the system prompt, the memory, the screenshot and the Second Instruction, followed by a standardized query prompt. By providing the agent with the "thought" from the corresponding step of the first interaction, we simulate a scenario where the agent "recalls" the user's specific habits or procedural preferences while observing the current state of the screen.

To isolate the agent's capability at each decision point and prevent the accumulation of errors typical in long-horizon tasks, we adopt a step-wise independent evaluation strategy. Rather than allowing the agent to proceed based on its own potentially erroneous prior actions, we provide the ground-truth visual state and the corresponding memory for every individual step. The evaluation framework then compares the agent's predicted action against the ground-truth action stored in the PERCU dataset. For interactive actions such as click, right-click, or double-click, which involve precise screen coordinates (x, y), we implement a coordinate parsing logic with a spatial tolerance. Specifically, we calculate the Euclidean distance between the predicted coordinates and the gold-standard coordinates. An action is categorized as correct if this distance is within a 50-pixel threshold. This approach acknowledges the inherent variability in visual grounding while ensuring the functional validity of the operation. By treating short-term memory as an in-context prompt component, our framework provides a lightweight yet effective mechanism for assessing personalized retrieval without the need for extensive model fine-tuning or complex vector database infrastructures.

C Details of Experimental Results for Routine Tasks on PERCU

While Section 6 primarily investigates agent capabilities on preference tasks, this section extends our evaluation to routine tasks, where agents must leverage historical context to execute generalized, multi-step procedural workflows. The experimental results for routine tasks are presented in Table 5.

Consistent with the preference tasks, GUI-Owl-7B (Ye et al., 2025) and GUI-R1-7B (Luo et al., 2025) maintain their positions as the leading architectures. GUI-Owl-7B (Ye et al., 2025) achieves the highest peak performance with a macro average accuracy of 29.29% and a micro average accuracy of 29.19%. Conversely, OS-Atlas-Pro-7B (Wu et al., 2024) continues to exhibit severe limitations in utilizing personalized context, with macro accuracies generally remaining below 8%. When

You are a helpful assistant who can help users complete computer tasks, with **full permission** to make any operations on the user's computer. Based on the provided current state, you need to suggest the next action to complete the task. Do not try to complete the entire task in one step. Break it down into smaller steps, and at each step you will get a new state to interact with.

You will receive:
1. MEMORY CONTEXT: The trajectory, habits, or rules extracted from your past operations on similar tasks for this user. You MUST use this memory to understand ambiguous requests in the current step.
2. CURRENT REQUEST: The instruction you need to fulfill right now.
3. A screenshot of the current state.

IMPORTANT: You must strictly adhere to the following rules: 1. Choose ONLY ONE action from the list below for each response, DO NOT perform more than one action per step. 2. Follow the exact syntax format for the selected action, DO NOT create or use any actions other than those listed. 3. Once the task is completed, output action finish.

Valid actions: 1. click (x, y) click the element at the position (x, y) on the screen. 2. right click (x, y) right click the element at the position (x, y) on the screen. 3. double click (x, y) double click the element at the position (x, y) on the screen. 4. drag from (x1, y1) to (x2, y2) drag the element from position (x1, y1) to (x2, y2). 5. scroll (x) scroll the screen vertically with pixel offset x. Positive values of x: scroll up, negative values of x: scroll down. 6. press key: key_content press the key key_content on the keyboard. 7. hotkey (key1, key2) press the hotkey composed of key1 and key2. 8. hotkey (key1, key2, key3) press the hotkey composed of key1, key2, and key3. 9. type text: text_content type content text_content on the keyboard. 10. wait wait for some time, usually for the system to respond, screen to refresh, advertisement to finish. 11. finish indicating that the task has been completed. 12. fail indicating that the task has failed, of this task is infeasible because not enough information is provided.

Response Format: Thought: {Your analysis combining the MEMORY CONTEXT and current screenshot}n\nAction: {The specific action you choose to take}

Figure 4: System Prompt for PERCU.

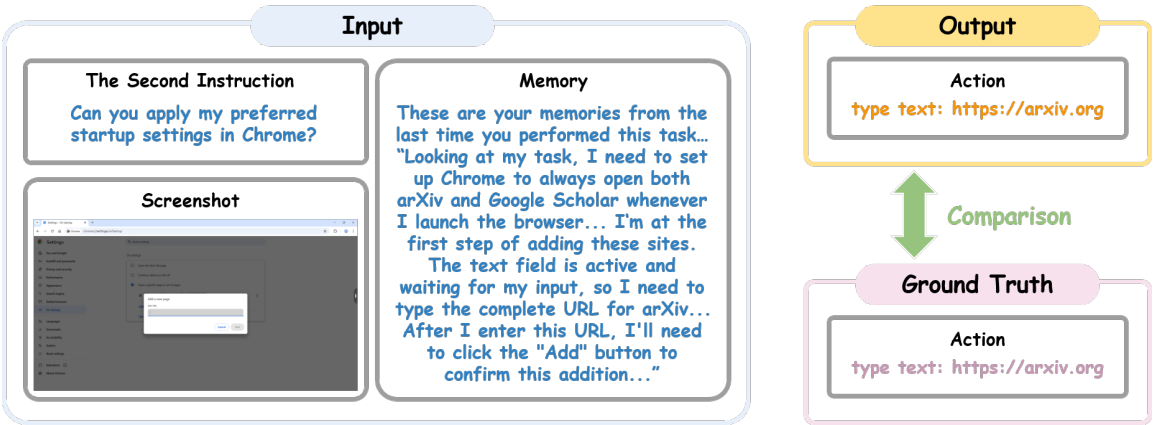

Figure 5: An example that demonstrates the evaluation process of PERCU.

examining the memory mechanisms within routine tasks, we observe the following trends. As for Information Distillation and Filtering, Compression and Consolidation Memory remains the most effective strategy for the smaller GUI-R1-3B (Luo et al., 2025) model, lifting its macro accuracy to 18.08% by filtering out verbose navigational noise and retaining high-density core assertions. Within the Relational Reasoning and Structuring dimension, Graph-Based Memory performs surprisingly poorly across all models (e.g., dropping to 16.28% for GUI-Owl-7B (Ye et al., 2025) and 12.62% for UI-TARS-1.5-7B (Qin et al., 2025b)). This suggests that constructing multi-hop relational entities might introduce excessive noise or over-complicate the context, disrupting the agent's step-by-step GUI grounding capabilities. Under the Execution Constraints and Retrieval dimension, Sequential Memory consistently yields the highest or near-highest performance across all leading models. This underscores that for routine tasks, preserving the exact chronological sequence of previous interactions is critical for the agent to deduce the correct procedural workflow.

By cross-referencing the results from preference tasks and routine tasks, several profound insights emerge regarding how agents handle different types of personalized contexts. There is a noticeable performance degradation when shifting from preference tasks to routine tasks. For instance, the peak macro accuracy of GUI-Owl-7B (Ye et al., 2025) drops from 35.99% in preference tasks to 29.29% in routine tasks. This disparity indicates that retrieving and applying a static user preference (e.g., "always use dark mode") is fundamentally easier for current MLLMs than generalizing a dynamic, multi-step user routine (e.g., "the standard procedure for filing a weekly report"). Despite the performance gap, Sequential Memory and OS-Like Memory prove to be universally robust architectures across both task types. This consistent trend highlights that whether an agent is recalling a personal aesthetic choice or a complex operational habit, strict chronological replay and structured context paging are the most reliable methods for memory integration in GUI environments.

D Boundaries of Personalized Capabilities

The advancement of personalized agents introduces a fundamental tension between effective generalization and over-presumptive reasoning, which directly impacts the safety and practical utility of the system. While the goal is for an agent to infer more from less, excessive extrapolation can lead to subjective hubris, where the agent violates user trust by acting on unverified assumptions. For instance, if an agent knows a user prefers a blue software theme, assuming the user also desires a blue desktop wallpaper might be a reasonable inference, but purchasing a blue physical cup based on the same signal constitutes a severe over-generalization. We posit that a safe and assistive personalized agent should operate within clearly defined boundaries. To facilitate future research, we categorize personalized capabilities into a four-level hierarchy, ranging from rote execution to prohibited extrapolation.

At the foundational level, Level 0 (L0) represents **Rote Memorization**, where the agent accurately reproduces learned knowledge in an identical context. This is exemplified by preference tasks where the agent recalls a specific entity, such as setting a favorite picture after being previously shown it. Building upon this, Level 1 (L1), defined as **Procedural Generalization**, constitutes the most significant practical value for desktop agents. In this stage, the agent applies an acquired workflow to a novel instance, such as generalizing a specific file extraction and organization routine from one project to another. Importantly, L1 focuses on the generalization of objective workflows rather than subjective whims, ensuring that the agent remains a reliable executor of the user's procedural routines. We argue that maximizing L1 capabilities while maintaining L0 stability should be the primary objective of current personalized computer use research.

In contrast, higher levels of generalization require significantly more caution. Level 2 (L2), defined as **Domain-Specific Interpolation**, involves applying a learned preference to a highly similar or related digital domain. An example would be an agent inferring a preference for dark mode in a code editor based on the user's consistent choice of dark themes in the operating system. While logically sound, L2 necessitates a degree of uncertainty management to avoid minor user friction. The most critical boundary, however, is Level 3 (L3) which is defined as **Cross-Domain Extrapolation**. This occurs when an agent takes a preference from a digital domain and projects it onto an unrelated physical or financial domain, such as unauthorized purchasing decisions. Following the safety principles, such behavior is categorized as a failure of the agent's alignment with user intent and must be strictly prohibited.

Ultimately, the hallmark of a sophisticated personalized agent lies not in its ability to guess correctly, but in its ability to recognize the limits of its memory. For scenarios falling into L2 or L3 where the personalized requirement is underspecified or uncertain, the personalized response should not be unilateral action, but rather proactive, memory-augmented inquiry. For example, instead of presumptively buying a blue cup, a reliable agent should state: "I noticed you prefer blue in your software themes. Does this preference extend to the item you wish to purchase, or would you like to see all available options?". By maximizing procedural generalization while strictly constraining cross-domain hallucinations through proactive clarification, we can develop agents that are both deeply personalized and fundamentally safe.

Memory Mechanisms	GUI-R1-3B		GUI-R1-7B		GUI-Owl-7B		UI-TARS-1.5-7B		OS-Atlas-Pro-7B	
	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro
Information Distillation and Filtering										
Sliding Window Memory	14.25	13.27	23.31	22.35	27.54	27.24	24.01	22.52	5.58	4.89
Summarization Memory	16.49	15.27	23.75	22.52	26.79	25.06	21.05	18.81	6.37	4.78
Compression and Consolidation Memory	18.08	15.63	23.54	21.76	20.32	18.34	20.53	17.87	5.23	3.83
Relational Reasoning and Structuring										
Hierarchical Memory	16.14	14.74	22.55	21.46	28.27	27.59	21.75	19.63	8.06	6.37
Graph-Based Memory	13.61	11.49	19.21	17.63	16.28	15.74	12.62	10.26	4.34	3.13
OS-Like Memory	14.86	12.91	24.04	22.99	28.11	27.71	21.61	18.99	6.93	5.37
Execution Constraints and Retrieval										
Sequential Memory	17.86	15.98	25.57	24.06	29.29	29.19	21.82	19.75	6.92	5.78
Retrieval-Based Memory	12.86	11.44	23.48	21.93	15.35	14.68	20.87	19.34	5.45	4.07
Memory-Augmented Memory	16.95	15.45	22.67	21.23	27.53	26.77	22.12	20.58	3.44	2.29

Table 5: Main results of routine tasks across different memory architectures on PERCU. "Macro" and "Micro" denote Macro Average Accuracy (%) and Micro Average Accuracy (%), respectively. Higher values indicate better performance.

E Future Development of Personalization

The path toward achieving true personalization necessitates that multimodal agents move beyond reactive execution to proactive memory utilization. Our research highlights that for an agent to master personalized capabilities, it must successfully bridge the semantic gap between underspecified subsequent instructions (Second Instructions) and the comprehensive short-term memory of initial interactions (First Instructions). Specifically, for preference tasks, the agent must establish a reliable mapping from abstract user concepts to concrete digital instances, such as grounding the favorite picture in a specific file like "Titanic.jpg". For routine tasks, the agent must internalize the transformation from high-level goals to idiosyncratic procedural workflows, ensuring that repetitive actions follow the user's unique organizational logic.

By formalizing these cognitive mappings, PERCU facilitates the evolution of multimodal agents from cold executors of detailed commands into intelligent partners capable of deciphering latent personalized intent. This transition is fundamental to realizing the vision of agents that act as seamless extensions of their users' digital lives. As the field moves toward more complex, long-horizon interactions, the ability to synthesize long-term episodic memory with real-time GUI observation will remain the defining characteristic of truly assistive technology. We hope that PERCU will provide the community with both the metric and the diagnostic insights necessary to drive the development of agents that are not only capable but also deeply attuned to the individual users they serve. However, while PERCU effectively assesses the agent's ability to utilize localized personalized knowledge for specific tasks, it does not yet address the challenges of autonomous memory management over extremely long horizons. In our current evaluation script, memory is injected as a fixed prompt to isolate the agent's reasoning capabilities at each decision point. However, in real-world deployments, agents must autonomously navigate vast memory hierarchies and resolve conflicts between competing or outdated historical priors without the benefit of ground-truth memory injection. Future iterations of PERCU could explore the integration of retrieval-augmented generation (Lewis et al., 2020; Gao et al., 2023; Jiang et al., 2023; Salemi and Zamani, 2024) or long context comprehension techniques (Ding et al., 2020; Manakul and Gales, 2021; Gao et al., 2021; Ivgi et al., 2023; Ratner et al., 2023) to evaluate how agents handle personalized knowledge that exceeds their immediate context window (Packer et al., 2023). Addressing these bottlenecks will be essential for transitioning from short-term personalized assistance to truly persistent and scalable digital companionship.

Looking toward the future development of personalized multimodal agents, our findings highlight the urgent need for advanced memory management mechanisms. Current architectures often treat historical data as a static, flat context, which becomes

problematic when faced with noisy or evolving user habits. A robust agent must incorporate a forgetting mechanism capable of distinguishing between obsolete long-term habits and the current working memory. For example, if a user has utilized ten different naming conventions in the past but has transitioned to a new project-specific format, the agent must possess the cognitive flexibility to suppress stale information in favor of recent contextual cues. Future research should thus focus on dynamic memory weighing and selective recall, ensuring that agents can maintain high personalization accuracy even in the presence of conflicting distractors in the user's history.