REAL-TIME PROCEDURAL LEARNING FROM EXPERIENCE FOR AI AGENTS

Dasheng Bi[*], Yubin Hu[*], Mohammed N. Nasir
Altrina
Menlo Park, California, USA
{dbi, harvey, mo}@altrina.com

ABSTRACT

Learning how to do things from trial and error in real time is a hallmark of biological intelligence, yet most LLM-based agents lack mechanisms to acquire procedural knowledge after deployment. We propose Procedural Recall for Agents with eXperiences Indexed by State (PRAXIS), a lightweight post-training learning mechanism that stores the consequences of actions and retrieves them by jointly matching environmental and internal states of past episodes to the current state. PRAXIS augments agentic action selection with retrieved state-action-result exemplars that are generated in real time. When evaluated on the REAL web browsing benchmark, PRAXIS improves task completion accuracy, reliability, and cost efficiency across different foundation model backbones, and shows preliminary generalization to unseen tasks in similar environments. These results demonstrate that PRAXIS enables the practical adoption of AI agents in fast-evolving stateful environments by helping them learn new procedures effectively.

1 INTRODUCTION

1.1 AI AGENTS AND PROCEDURAL LEARNING

AI agents are artificial intelligence systems capable of observing and taking actions in an environment. As adoption spreads across industries, there is a growing need for agents to quickly learn domain- or user-specific information. There are two main classes of information that agents are typically expected to learn:

1. **Facts**: atomic pieces of information independent of the state of the agent or environment (*e.g.*, a user's name). Facts are generally context-independent.
2. **Procedures**: conventions for doing things (*e.g.*, "how to troubleshoot a failed login"). Procedures are a sequence of state-dependent requirements or preferences over actions.

In real-world applications, learning and optimizing procedures in real time are at least as important as learning facts. While frameworks such as Mem0 (Chhikara et al., 2025) focus on long-term *factual* memory, efficient learning of *procedures* in AI agents remains relatively underexplored.

A naïve approach is specifying standard operating procedures (SOPs) that are included in the agent's context, effectively reducing procedures to a large bundle of facts. In practice, this approach faces challenges: (1) many procedures are not documented, as humans are often trained via demonstration; (2) enumerating all states and edge cases in a combinatorial space is difficult; and (3) procedures become obsolete quickly as environments change. We argue that a more robust approach is to learn from demonstrations or experiences directly. Inspired by state-dependent memory in psychology (Tulving & Thomson, 1973), we propose PRAXIS, a method for procedural learning compatible with both demonstrations by human experts and experiences generated by an AI agent.

[*]Both authors contributed equally to this research.

1.2 WEB AGENTS AND THE BROWSER ENVIRONMENT

Human-facing web applications almost always require multi-step interactions to accomplish meaningful goals (*e.g.*, purchasing an item online requires searching, filtering, logging in, completing forms, and checking out)—a procedure. These procedures must also adapt to changing environments (*e.g.*, an e-commerce site may have seasonal pop-ups or redesigned interfaces), making web browsing a natural environment to study procedural learning. As the tasks are obvious to humans, comprehensive procedures are rarely documented. This lack of data and high personalization requirements limit pretraining coverage in foundation models. A post-training, state-indexed procedural memory thus becomes essential for robust web automation, allowing agents to acquire and reuse procedures as new states appear.

2 RELATED WORK

External Memory for AI Chatbots A broad class of systems augment LLMs with non-parametric memory in the conversational environment. Retrieval-augmented generation (RAG) attaches a document store to provide factual knowledge at inference time (Lewis et al., 2021). In agentic settings, mem0 (Chhikara et al., 2025) provides a queryable, cross-session memory for user preferences and long-range conversational context. Academic frameworks include MemoryBank (Zhong et al., 2023) which mimics human long-term memory with continual decay and reinforcement. These approaches focus on factual memory for agents in conversational environments. In contrast, our method focuses on learning action policies in stateful visual environments that are significantly more complex and not entirely observable like the web environment.

Experience-Based Self-Improvement and Workflow Memories A complementary line of work improves agents via self-reflection. Reflexion (Shinn et al., 2023) maintains verbal reflections in an episodic buffer to guide subsequent trials; and CLIN (Majumder et al., 2023) performs continual task adaptation with a persistent textual memory of causal abstractions. These methods have not been extensively-tested in visual environments and generally do not encode information of environmental state. Agent Workflow Memory (Wang et al., 2024) and Synapse (Zheng et al., 2024) induce abstracted, natural language workflows from successful trajectories and retrieve them to augment prompts at test time. In contrast, our method performs local state-based recall that is grounded primarily in the live environment state, a factor not present in prior works, and secondarily to the goal. Moreover, we index memories with explicit state and action descriptors, enabling precise recall and learning of minute details required for environments like the web.

3 METHODS

3.1 ALTRINA AGENT

The experiments in this study were conducted with Altrina[1], a frontier AI agent operating in computer use environments. Altrina is capable of perceiving the environment both visually and as compressed textual information, taking actions on a computer, and executing complex directives end-to-end. We restrict the action space to web-related actions for this study. In this environment, we previously showed that the baseline Altrina agent achieves state-of-the-art results on multiple benchmarks (Altrina (formerly Tessa AI), 2025a;b), including WebVoyager (He et al., 2024) and REAL (Garg et al., 2025). Altrina's scaffolding layer orchestrates underlying foundation models with a "node-based" architecture. Each node is designed for a specific function in the agentic loop. In this study, we modify the *action selection node*, which is responsible for deciding what the next agent action should be given the current state and progress towards the task objective, by adding a dedicated *procedural memory* section to its context.

[1]The agent system described in this work was previously known as Tessa in prior publications (Altrina (formerly Tessa AI), 2025a;b).

3.2 State-Dependent Memory

Mirroring state-dependent memory in psychology, which finds improved recall when internal state and external context at retrieval match those present during encoding (Tulving & Thomson, 1973), we designed a state-dependent memory for Altrina in which indexing and retrieval are based on the browser's environment state and the agent's internal state. Each memory entry contains the following components:

1. $M_i^{\text{env-pre}}$, a description of the environmental state in which this memory is generated
2. M_i^{int}, the internal state of the agent at the time of the experience, including the overall directive the agent is trying to achieve
3. a_i, the action taken at the time
4. $M_i^{\text{env-post}}$, the state of the environment after the action was taken

Formally, we use the following procedure (Alg. 1) to retrieve a set of memories:

Algorithm 1: Procedural Memory Retrieval

Input: Memory environment states $\{M_i^{\text{env}}\}_{i=1}^n$, query environment state Q^{env}, memory internal states $\{M_i^{\text{int}}\}_{i=1}^n$, query internal state Q^{int}, internal state embedding function f, search breadth k, similarity threshold τ

Output: Retrieved memory indices $\mathcal{R}$

for $i \leftarrow 1$ **to** n **do**

$v_i \leftarrow \textsc{IoU}(M_i^{\text{env}}, Q^{\text{env}})$;

$\ell_m \leftarrow |M_i^{\text{env}}|,\ \ell_q \leftarrow |Q^{\text{env}}|$;

$$l_i \leftarrow \textsc{LengthOverlap}(\ell_m, \ell_q) = 1 - \frac{|\ell_m - \ell_q|}{\max(\ell_m, \ell_q)};$$

$s_i^{\text{env}} \leftarrow v_i \cdot l_i$;

$s_i^{\text{int}} \leftarrow \langle f(M_i^{\text{int}}), f(Q^{\text{int}}) \rangle$;

$\mathcal{R}^{\text{env}} \leftarrow \textsc{TopkIndices}(s^{\text{env}}, k)$;

$\widetilde{\mathcal{R}} \leftarrow \textsc{Sorted}(\mathcal{R}^{\text{env}};\ \text{key} = i \mapsto s_i^{\text{int}};\ \text{descending})$;

$\mathcal{R} \leftarrow [\, i \in \widetilde{\mathcal{R}} \mid s_i^{\text{env}} \geq \tau \,]$;

return $\mathcal{R}$;

3.3 Benchmark

Web agent benchmarks Mind2Web and WebVoyager target relatively straightforward workflows on live websites (Deng et al., 2023; He et al., 2024); WebArena hosts replicas of a few websites for local testing of relatively simple tasks with text-based evaluations (Zhou et al., 2024). To ensure reproducible evaluation on tasks of real-world relevance, we evaluate on deterministic clones of functional websites using the REAL benchmark (Garg et al., 2025). REAL provides replicas of 11 commonly used sites and 112 everyday tasks of varying complexity, along with both programmatic state checks for action tasks and rubric-guided natural language evaluations for information retrieval tasks. Prior benchmarking reports that frontier models with naïve scaffolding achieve at most $\sim 41\%$ success, leaving substantial room for post-training procedural learning.

4 Results

4.1 Procedural memory improves agent accuracy

We benchmarked Altrina on REAL using several VLM backbones for each of the agentic compute nodes, either with or without access to procedural memory (Fig. 1). Procedural memory consistently improved the performance of Altrina (average performance across models increased from 40.3% to 44.1%, mean accuracy over five repetitions).

We also measured the best-of-5 accuracy (Table 1), which reflects the agent's capability frontier under repeated attempts. Procedural memory also improved best-of-5 accuracy (average across models from 53.7% to 55.7%). Together, these results suggest that retrieved procedural memory traces provide reusable priors that improve agentic behavior on complex tasks, and that such memory traces may be generalized across similar tasks, supporting the completion of novel tasks.

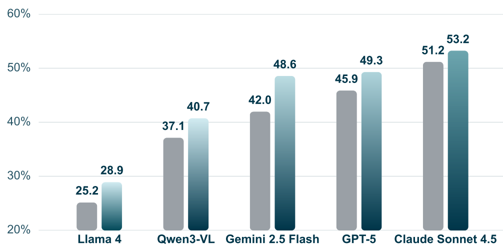

Figure 1: Performance on REAL benchmark with (blue) and without (grey) PRAXIS

Model	Base	With PM
Llama 4	47.3	**52.7**
Qwen3-VL	44.6	**47.3**
Gemini 2.5 Flash	59.8	**61.6**
GPT-5	56.2	**57.1**
Claude Sonnet 4.5	**60.7**	59.8

Table 1: Best-of-5 accuracy over tasks.

4.2 Procedural memory improves agent reliability

We define *reliability* as the mean success rate over five repetitions of a task, averaged over all tasks in REAL having at least one successful run over the five repetitions. Procedural memory improved the reliability of Altrina from 74.5% to 79.0% averaged across all models (Fig. 2). These results suggest that retrieved procedural memory traces can suppress unwanted stochastic variance in the underlying vision-language models by biasing their decisions towards previously successful trajectories under similar states, thereby improving the reliability and repeatability of the high-level agentic behavior.

4.3 Procedural memory improves agent efficiency

We measure *efficiency* as the average number of steps taken on tasks in REAL with at least one successful run across five repetitions. Procedural memory reduces steps-to-completion on average from 25.2 to 20.2 across models (Table 2). This indicates that procedural memory recall effectively steers agentic behavior along both correct and more direct trajectories, avoiding common pitfalls.

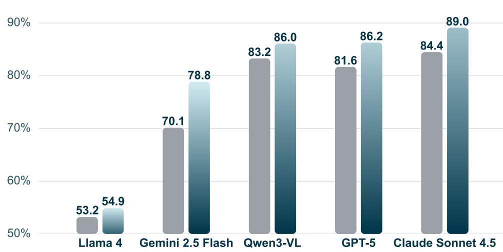

Figure 2: Reliability on REAL benchmark with (blue) and without (grey) procedural memory.

Model	Base	With PM
Llama 4	19.8	**16.2**
Qwen3-VL	27.7	**20.8**
Gemini 2.5 Flash	28.9	**22.3**
GPT-5	24.2	**20.7**
Claude Sonnet 4.5	25.2	**21.0**

Table 2: Average number of steps to complete task.

4.4 Procedural memory performance scales with retrieval breadth

We ablated the retrieval breadth hyperparameter k in Alg. 1 to study its effect on the performance of procedural memory (Fig. 3) with a Gemini 2.5 Flash backbone. We found that performance generally increased as we broadened our retrieval breadth and converged to a plateau, suggesting that while there exists some local context crowding, procedural memory offers increasingly helpful and generalizable context to the language model at a larger scale.

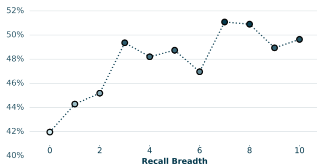

Figure 3: Performance vs. retrieval breadth.

5 Discussion

As AI agents increasingly work alongside humans, methods like PRAXIS, which enable real capability customization with user data and procedures while preserving privacy, will be critical to agent adoption. Our results demonstrate that state-dependent procedural memory provides reusable state-to-action priors that improve accuracy, reliability, and efficiency across diverse VLM backbones. Future directions include extending to general computer use, richer state encoders for improved retrieval, adaptive mechanisms that account for uncertainty, and learning from user feedback to encode preferences for how tasks are performed.

References

Altrina (formerly Tessa AI). Introducing large neurosymbolic cognitive models. `https://www.altrina.com/blog/introducing-large-neurosymbolic-cognitive-models`, 2025a. Accessed: 2025-11-16.

Altrina (formerly Tessa AI). Evolving our state-of-the-art browsing agent. `https://www.altrina.com/blog/evolving-our-state-of-the-art-browsing-agent`, 2025b. Accessed: 2025-11-16.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory, April 2025. URL `http://arxiv.org/abs/2504.19413`. arXiv:2504.19413 [cs].

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a Generalist Agent for the Web, December 2023. URL `http://arxiv.org/abs/2306.06070`. arXiv:2306.06070 [cs].

Divyansh Garg, Shaun VanWeelden, Diego Caples, Andis Draguns, Nikil Ravi, Pranav Putta, Naman Garg, Tomas Abraham, Michael Lara, Federico Lopez, James Liu, Atharva Gundawar, Prannay Hebbar, Youngchul Joo, Jindong Gu, Charles London, Christian Schroeder de Witt, and Sumeet Motwani. REAL: Benchmarking Autonomous Agents on Deterministic Simulations of Real Websites, April 2025. URL `http://arxiv.org/abs/2504.11543`. arXiv:2504.11543 [cs].

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an End-to-End Web Agent with Large Multimodal Models, June 2024. URL `http://arxiv.org/abs/2401.13919`. arXiv:2401.13919 [cs].

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, April 2021. URL `http://arxiv.org/abs/2005.11401`. arXiv:2005.11401 [cs].

Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Peter Jansen, Oyvind Tafjord, Niket Tandon, Li Zhang, Chris Callison-Burch, and Peter Clark. CLIN: A Continually Learning Language Agent for Rapid Task Adaptation and Generalization, October 2023. URL `http://arxiv.org/abs/2310.10134`. arXiv:2310.10134 [cs].

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language Agents with Verbal Reinforcement Learning, October 2023. URL `http://arxiv.org/abs/2303.11366`. arXiv:2303.11366 [cs].

Endel Tulving and Donald M. Thomson. Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80(5):352–373, 1973. ISSN 1939-1471. doi: 10.1037/h0020071. Place: US Publisher: American Psychological Association.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent Workflow Memory, September 2024. URL `http://arxiv.org/abs/2409.07429`. arXiv:2409.07429 [cs].

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-Exemplar Prompting with Memory for Computer Control, January 2024. URL http://arxiv.org/abs/2306.07863. arXiv:2306.07863 [cs].

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing Large Language Models with Long-Term Memory, May 2023. URL http://arxiv.org/abs/2305.10250. arXiv:2305.10250 [cs].

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A Realistic Web Environment for Building Autonomous Agents, April 2024. URL http://arxiv.org/abs/2307.13854. arXiv:2307.13854 [cs] version: 4.