PLAN-AWARE AUTOMATED CONTEXT ENGINEERING

Kamer Ali Yuksel & Hassan Sawaf
aiXplain Inc., San Jose, CA, USA
{kamer, hassan@aixplain.com}

ABSTRACT

Associative memory has re-emerged as a central abstraction for understanding attention, retrieval, and state evolution in modern AI systems, particularly in memory-augmented and agentic models. In large language model (LLM) agents, memory is instantiated as an evolving prompt context comprising plans, intermediate reasoning, tool outputs, retrieved documents, and instructions. As agents execute long-horizon workflows, this memory state grows rapidly, becoming noisy, unstable, and increasingly difficult to reason over. We introduce **PAACE** (**P**lan-**A**ware **A**utomated **C**ontext **E**ngineering), a framework that formulates context management as a problem of *associative memory shaping*. PAACE learns to selectively retain, rewrite, compress, or discard memory elements based on their associative relevance to plan steps, effectively stabilizing task-relevant memory attractors while suppressing irrelevant or distracting states. Unlike query-aware or single-step compression methods, PAACE explicitly conditions memory transformations on the next-k tasks in an agent's plan, enabling multi-step associative retrieval and long-horizon reasoning. PAACE consists of two components: (1) **PAACE-Syn**, a scalable generator of synthetic agent workflows with explicit plan structure and stepwise memory supervision, and (2) **PAACE-FT**, a family of compact, distilled associative memory operators trained to imitate successful teacher-guided memory transformations. Experiments on OfficeBench and multi-objective QA demonstrate that PAACE improves agent accuracy while substantially reducing memory load and cumulative attention cost. We show that learned associative memory shaping not only improves efficiency but also acts as a form of regularization, stabilizing reasoning over extended interactions.

1 INTRODUCTION

Associative memory (AM) has long served as a foundational concept in cognitive science, neuroscience, and machine learning, offering a principled view of how systems retrieve, stabilize, and manipulate information through similarity, relevance, and attractor dynamics. Recent work has revealed deep connections between associative memory and modern deep learning components, including attention mechanisms, energy-based models, and retrieval-augmented architectures. In parallel, large language model (LLM) agents have emerged as a dominant paradigm for multi-step reasoning, planning, and tool use across diverse domains. In LLM agents, memory is not encoded in persistent weights but instead materializes as an *explicit, mutable context*: system instructions, plans, reasoning traces, tool outputs, retrieved documents, and long-term summaries. This context functions as the agent's working memory and state representation. As agents execute long-horizon workflows, this memory grows rapidly and becomes increasingly noisy, redundant, and unstable. Even models with very large context windows suffer from attention dilution and reasoning failures when exposed to poorly structured or weakly relevant memory states.

An LLM agent's evolving context can be viewed as a high-dimensional memory state in which attention implements soft associative retrieval. Failures in long-horizon reasoning are thus often failures of associative stability: weakly related or stale memory fragments form spurious attractors that compete for attention, interfering with task-relevant recall and state evolution. We argue that many failures of long-horizon agents are fundamentally *associative memory failures*. Irrelevant or stale memory fragments continue to associate with the current state, competing for attention and destabilizing reasoning. Conversely, task-critical information may be buried or inadvertently discarded. Existing approaches to context management—such as summarization, retrieval, or heuristic prun-

ing—address this problem only partially. Most methods are query-aware rather than plan-aware, operate at a single step, or lack mechanisms for jointly refining memory content and instructions.

In this work, we introduce **PAACE**, a framework that explicitly treats context management as *learned associative memory shaping*. PAACE learns how to transform an agent's memory state at each step by conditioning on plan structure, thereby reinforcing memory elements that are associatively relevant to upcoming tasks and suppressing those that are not. Rather than relying on fixed heuristics or architectural changes, PAACE learns memory transformation policies from data, using large-scale synthetic agent workflows and teacher-guided supervision. Our contributions connect associative memory principles with practical agentic systems, offering a concrete instantiation of memory shaping, attractor stabilization, and attention control in long-horizon LLM agents:

1. We formulate context management in agents as a problem of **associative memory shaping**, linking agent memory to relevance-driven stabilization and suppression of memory states.
2. We introduce **PAACE**, the first framework to perform *plan-aware, next-k associative memory optimization* through learned context transformation policies.
3. We propose **PAACE-Syn**, a scalable synthetic workflow generator that produces long-horizon agent trajectories with explicit plan structure and dense memory supervision.
4. We present **PAACE-FT**, a family of compact, distilled associative memory operators that approximate teacher-guided memory shaping at low inference cost.
5. We demonstrate empirically that associative memory shaping improves both accuracy and efficiency across multiple long-horizon agent benchmarks.

2 BACKGROUND

In LLM agents, memory is not realized as persistent neural weights or fixed external buffers, but as an explicit, evolving *context* composed of instructions, plans, intermediate reasoning, tool outputs, retrieved documents, and summaries. This context functions as a high-dimensional associative memory state: at each step, attention performs soft associative retrieval over heterogeneous memory fragments, weighting them by relevance to the current task. From this perspective, agent behavior emerges from repeated cycles of associative retrieval and state update over an expanding memory substrate. Failures in long-horizon agent reasoning are therefore often failures of associative memory. As interaction histories grow, weakly related, obsolete, or redundant fragments accumulate and compete for attention, leading to distraction, instability, and loss of task-relevant information. Even models with very large context windows exhibit degraded performance when associative interference overwhelms selective recall. This motivates treating context management not as bookkeeping, but as an explicit associative optimization problem. Classical associative memory models, such as Hopfield networks, formalize recall as convergence toward attractor states in an energy landscape (Hopfield, 1982). Modern variants, including dense and continuous Hopfield networks, substantially increase storage capacity and reveal a close correspondence between associative recall and attention mechanisms (Krotov & Hopfield, 2016; 2021; Ramsauer et al., 2020). In particular, Transformer attention can be interpreted as a form of dense associative retrieval, where memory items compete based on similarity and are softly aggregated to produce the next state. Recent theoretical work has further strengthened this connection by framing attention as implicit energy minimization or test-time optimization over memory states (Niu et al., 2024). These analyses highlight that retrieval stability depends not only on similarity scores; but also on how memory states are retained, suppressed, or reshaped over time. While these models operate at the neural activation level, they provide a conceptual foundation for understanding LLM agent context-level associative dynamics.

Modern agent frameworks increasingly augment LLMs with explicit memory mechanisms to support long-horizon reasoning and tool use. Existing approaches typically rely on heuristic truncation, summarization, retrieval, or architectural memory modules. However, most methods are *query-aware* or *single-step* in nature: they optimize memory relevance with respect to the current query, without considering how memory will be used across future steps in a plan. As a result, information required later may be prematurely discarded, while irrelevant fragments persist. Recent work has shown that agents can learn to construct and update memory through reinforcement or supervision Wang et al. (2025), reinforcing the view that memory is an actively optimized substrate rather than a passive log. Nonetheless, these systems largely focus on what to store, not how to continuously

reshape existing context to preserve multi-step dependencies and stabilize reasoning trajectories. PAACE builds on these insights by treating agent context as an associative memory state that can be *reshaped at test time*. Rather than introducing new architectural memory modules or explicit energy functions, PAACE operates directly on the symbolic–textual context used by the agent. Conditioning memory transformations on upcoming plan steps induces a form of multi-step associative relevance: memory fragments that consistently support future tasks are preserved or rewritten into compact representations, while weakly coupled or stale fragments are suppressed. This view aligns with modern interpretations of associative memory as test-time optimization and attention control, rather than static recall. PAACE complements classical and modern associative memory models by operating at the level of explicit context, enabling integration with any backbone model or agent framework; and provides a practical instantiation of associative memory shaping for long-horizon agents, directly addressing the instability and inefficiency observed in unconstrained context growth.

3 METHODOLOGY

From an associative memory viewpoint, an LLM agent's context can be seen as a high-dimensional memory state composed of heterogeneous fragments: instructions, symbolic variables, textual observations, retrieved knowledge, and intermediate reasoning. At each step, the agent performs associative retrieval over this memory via attention, implicitly weighting fragments by relevance to the current task. However, without intervention, this memory state evolves monotonically, accumulating weakly associated or obsolete fragments. This leads to what has been described as *context rot*: a degradation of effective associative retrieval due to excessive competition among memory elements. PAACE addresses this by explicitly reshaping the memory landscape. Conditioning on plan steps induces a form of multi-step associative relevance, analogous to stabilizing task-relevant attractors in memory space. Memory elements that are repeatedly useful across upcoming tasks are preserved or rewritten into compact representations, while weakly associated elements are suppressed. Although PAACE does not implement an explicit energy function, its learned transformations implicitly encourage low-entropy, stable memory states that support consistent reasoning across long horizons. While PAACE does not instantiate an explicit Hopfield-style energy function, its learned memory transformations can be interpreted as implicitly reshaping the agent's associative energy landscape: stabilizing memory configurations that consistently support upcoming plan steps while suppressing unstable or weakly coupled states. In this sense, PAACE operates as a learned, task-conditioned attractor-shaping mechanism at test time. This framing aligns PAACE with modern interpretations of associative memory as test-time optimization and attention control, rather than static recall. PAACE formulates context management in long-horizon LLM agents as a problem of *learned associative memory shaping*. Rather than treating the agent's context as an ever-growing log, PAACE learns to transform the agent's memory state at each step by selectively retaining, rewriting, compressing, or discarding memory fragments based on their relevance to plan steps.

The framework follows a two-stage design:

1. a high-capacity **teacher** LLM that performs plan-aware, next-k–conditioned memory shaping using a learned natural-language policy;
2. a compact **student** model distilled to approximate the teacher's associative memory transformations at low inference cost.

This allows PAACE to learn memory shaping policies directly from data, without requiring handcrafted relevance rules or architectural changes. At step t, an agent maintains an explicit memory:

$$C_t = \{I_0,\ P,\ \Pi,\ H_{0:t},\ O_{0:t},\ R_{0:t},\ M\}, \tag{1}$$

where I_0 is the initial user instruction, P the system prompt, $\Pi = [\tau_1, \ldots, \tau_n]$ the plan, $H_{0:t}$ reasoning traces, $O_{0:t}$ tool or environment observations, $R_{0:t}$ retrieved documents, and M long-term memory. This heterogeneous collection functions as the agent's working memory and associative state. Uncontrolled growth of C_t leads to degraded associative retrieval: irrelevant or stale fragments compete for attention, destabilizing reasoning. PAACE addresses this by learning how to reshape C_t conditioned on future tasks. At each step, PAACE applies a learned transformation

$$\tilde{C}_t = \mathrm{TeacherCompress}(C_t,\ \Pi_{t:t+k};\ p), \tag{2}$$

where $\Pi_{t:t+k}$ denotes the next k plan steps and p is a learned natural-language compression policy. Conditioning on multiple future tasks induces a form of multi-step associative relevance: memory

Table 1: **OfficeBench and Multi-Objective QA results.** PAACE consistently improves task performance while reducing memory footprint and cumulative attention cost. Conditioning memory shaping on plan steps preserves cross-step dependencies more effectively than below baselines.

	OfficeBench (Wang et al., 2024)				**Multi-Objective QA (Zhou et al., 2025)**				
Method	Acc↑	Steps↓	Peak↓	Dep↓	EM↑	F1↑	Steps↓	Peak↓	Dep↓
No Compression	76.84	11.52	7.27	4.43	0.366	0.488	15.78	10.35	3.32
FIFO	67.37	12.26	4.02	2.64	0.293	0.388	19.26	5.09	2.51
Retrieval	65.26	16.20	4.33	2.06	0.331	0.438	20.06	5.11	2.62
LLMLingua	70.53	10.89	4.65	1.85	0.363	0.481	17.68	5.68	2.24
Prompting	71.58	10.13	4.40	1.10	0.376	0.478	18.70	4.73	1.66
ACon UT	74.74	13.13	4.93	3.85	0.373	0.494	17.14	4.71	1.57
ACon UTCO	72.63	11.54	4.54	1.91	0.335	0.458	17.79	4.65	1.50
PAACE (ours)	**78.10**	**10.48**	**4.29**	**1.64**	**0.402**	**0.512**	**16.86**	**4.41**	**1.41**

elements useful across upcoming tasks are preserved or rewritten, while weakly associated elements are suppressed. Importantly, PAACE does not rely on explicit symbolic relevance scores or energy functions. Instead, relevance is operationalized through *outcome preservation*: compressed memory states are accepted only if they preserve task performance over full execution trajectories.

To obtain dense supervision for associative memory shaping, PAACE generates a large corpus of synthetic agent workflows. Each workflow consists of a noisy initial input containing irrelevant and redundant information, an explicit multi-step plan Π, a sequence of task instructions $\{\tau_t\}$, a final output specification. Each workflow is executed twice: once with the full memory state and once with teacher-guided compression applied at every step. Let $\hat{y}^{\text{full}}$ and $\hat{y}^{\text{comp}}$ denote the final outputs. A compressed trajectory is considered *successful* if:

- semantic similarity between outputs exceeds a threshold,
- compression ratios are valid ($0 < |\tilde{C}_t|/|C_t| < 1$),
- an LLM-based evaluator does not judge the compressed outcome as worse.

Only successful trajectories are retained, yielding supervision pairs $(\Pi_{t:t+k},\, C_t) \rightarrow \tilde{C}_t$, which captures function-preserving associative memory transformations aligned with plan structure. The teacher's compression behavior is controlled by a natural-language prompt p, which is optimized via an LLM-driven evolutionary process. Prompt variants are evaluated across workflows and ranked using a composite score that balances: success rate, semantic fidelity, and compression strength. Over time, this procedure yields a robust associative memory shaping policy generalizing across tasks and domains. PAACE-FT (student) is trained to imitate the teacher's transformations using standard causal language modeling. Given input $x = (\Pi_{t:t+k}, C_t)$ and target $\tilde{C}_t$, it minimizes:

$$\mathcal{L} = -\mathbb{E}_{(x,y)} \sum_i \log p_\theta(y_i \mid x, y_{<i}), \tag{3}$$

masking input tokens from the loss. The resulting model performs fast, plan-aware associative memory shaping at inference time, enabling practical deployment in long-horizon agent loops.

4 Experiments

We evaluate PAACE on long-horizon agent benchmarks designed to stress memory management, multi-step reasoning, and tool interaction. Our goal is to assess whether learned associative memory shaping improves agent correctness while reducing memory load and cumulative attention cost. These benchmarks require agents to maintain coherent memory states across many steps and are sensitive to context overload. We compare PAACE against several context-management baselines, including a no-compression setting that preserves the full interaction history, FIFO truncation that retains only the most recent turns, embedding-based retrieval of past interactions, LLMLingua for extractive long-context compression, heuristic prompt-based summarization, and ACon with both UT and UTCO variants. Evaluation focuses on both task effectiveness and memory efficiency. We report benchmark-specific task performance metrics (Accuracy, Exact Match, or F1), the number of interaction steps required to complete each task, the peak context length observed during execution,

and the cumulative dependency $\sum_t |C_t|$, which approximates total attention cost and correlates with latency and quadratic transformer compute. The teacher compressor is implemented using a 120B-parameter LLM with a 65k-token context window, ensuring strong multi-step reasoning during supervision. The student PAACE-FT model is distilled into Qwen3-4B-Instruct. Across benchmarks, the student retains approximately 97–98% of teacher performance while reducing inference cost.

PAACE consistently improves task performance while substantially reducing memory usage. On OfficeBench and Multi-Objective QA, it improves accuracy and F1 while reducing memory cost and interaction steps. Notably, PAACE often outperforms the no-compression baseline, indicating that associative memory shaping acts as a form of regularization: removing weakly associated or stale memory fragments stabilizes reasoning and reduces distraction. Query-aware or heuristic compression methods optimize relevance for a single step, often discarding information needed later. In contrast, PAACE conditions memory transformations on multiple future tasks, preserving cross-step dependencies such as variable bindings, tool-call results, and intermediate constraints. This results in more coherent memory states that support stable long-horizon reasoning. These findings support the view that context failures in agentic systems are fundamentally associative memory failures. Explicit, learned memory shaping provides an effective solution.

5 Conclusion

We presented PAACE, a framework for plan-aware associative memory shaping in long-horizon LLM agents. By treating context as an explicit memory state and learning how to reshape it based on future task relevance, PAACE bridges classical ideas from associative memory with modern agentic AI systems. Our results demonstrate that learned memory shaping not only reduces computational cost but also stabilizes reasoning, acting as a form of regularization over extended interactions. This work highlights associative memory as a unifying abstraction for understanding attention, retrieval, and state evolution in agentic systems. We believe that future agent architectures will increasingly rely on explicit, learned memory operators such as PAACE, positioning associative memory not merely as a retrieval mechanism, but as a core substrate for intelligent behavior; which complements classical associative memory models such as Hopfield networks and dense associative memories by operating at the level of explicit symbolic–textual memory rather than neuron-level states, and also differs from architectural memory modules by treating associative memory as a test-time optimization problem over context, enabling integration with any backbone model or agent framework.

References

John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

Dmitry Krotov and John J. Hopfield. Dense associative memory for pattern recognition. *arXiv preprint arXiv:1606.01164*, 2016.

Dmitry Krotov and John J. Hopfield. Large associative memory problem in neurobiology and machine learning. *arXiv preprint arXiv:2008.06996*, 2021.

Xueyan Niu, Bo Bai, Lei Deng, and Wei Han. Beyond scaling laws: Understanding transformer performance with associative memory. *arXiv preprint arXiv:2405.08707*, 2024.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Yu Wang, Ryuichi Takanobu, Zhiqi Liang, Yuzhen Mao, Yuanzhe Hu, and Xiaojian Wu. Mem-α: Learning memory construction via reinforcement learning. *arXiv preprint arXiv:2509.25911*, 2025.

Zilong Wang, Yuedong Cui, Li Zhong, Zimin Zhang, Da Yin, Bill Yuchen Lin, and Jingbo Shang. Officebench: Benchmarking language agents across multiple applications for office automation. *arXiv preprint arXiv:2407.19056*, 2024.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Jinhua Zhao, Bryan Kian Hsiang Low, and Paul Pu Liang. MEM1: Learning to synergize memory and reasoning for efficient long-horizon agents. *arXiv preprint arXiv:2506.15841*, 2025.