Bridging the Semantic Gap in Long-Term Agent Memory Retrieval

Sixu Chen
Department of Computer Science
Kent State University
Kent, OH 44240, USA
schen53@kent.edu

Qiang Guan
Department of Computer Science
Kent State University
Kent, OH 44240, USA
qguan@kent.edu

***Abstract*—LLM-based agents perform well across many tasks but still struggle with long-term interactions requiring coherence, continuity, and personalization. LLM-based agents often use RAG-style memory systems to retrieve past experiences, but direct query–memory similarity can fail when user queries and fine-grained memories are semantically misaligned. We address this semantic gap with an iterative, anchor-guided retrieval strategy inspired by spreading activation in associative memory. Starting from a small set of seed memories, the method expands through global semantic associations and local temporal neighbors, while a path-aware cross-encoder selects complementary evidence across multiple hops. Retrieved memories are accumulated and reused as anchors for further exploration.**

***Index Terms*—LLM, Agent, Agent memory**

I. Introduction

Long-term LLM agents require memory to preserve continuity and personalization across interactions. Since limited context windows cannot retain growing interaction histories, external memory systems are used to store and retrieve past experiences [1]. Recent methods further improve memory organization through structured and dynamically updated representations [2].

Despite advances in memory construction, retrieval remains a major bottleneck. Most systems still follow a standard RAG pipeline: a query is compared with individual memories in embedding space, and the top-k most similar memories are directly provided to the LLM. This assumption is poorly matched to conversational agent memory. Memories are often short, episodic, temporally dependent, and semantically overlapping, while user questions may describe the same event from a different perspective or at a higher level of abstraction [3]. Consequently, the memory containing the answer may have only weak direct similarity to the query.

Our analysis on LoCoMo [4], a long-term conversational memory benchmark in Fig. 1 shows that many gold memories are absent from the top-10 retrieval results, and increasing the retrieval budget to top-50 still leaves substantial evidence unrecovered.

We explore memory retrieval through memory-to-memory associations rather than direct query-to-memory matching. Query-relevant memories serve as anchors, which are expanded through semantic and temporal neighbors. Candidates are then reranked using both the query and their parent anchor.

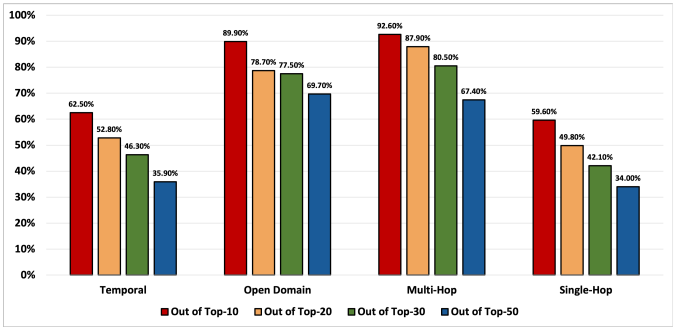


Fig. 1. Percentage of queries whose mean rank of gold memories falls outside Top-K under direct query–memory embedding retrieval on LoCoMo. Higher values indicate weaker evidence recall and highlight the semantic gap between queries and fine-grained agent memories.

Selected memories are accumulated as evidence and reused for multi-hop exploration, enabling recovery of relevant memories missed by one-shot retrieval while reducing noisy context.

II. Methodology

Our approach treats memory retrieval as an iterative exploration process rather than a one-shot query–memory search. Starting from a few query-relevant memories, the retriever follows semantic and temporal connections between memories to uncover evidence that may be difficult to access directly.

A. Structured Memory Representation

Each interaction is stored as an individual memory together with its timestamp. An LLM enriches the original content with keywords, semantic tags, and a short contextual description. These elements are jointly encoded to obtain a high-level memory representation for subsequent retrieval.

B. Anchor-Based Neighborhood Exploration

For each query, we first retrieve a small number of relevant memories as **seed anchors**. Instead of directly using these memories as the final context, we expand around each anchor through two complementary neighborhoods:

- **Semantic neighbors:** memories closely related to the anchor in embedding space, enabling exploration of globally associated experiences.

TABLE I
RESULTS ON GVD USING LLAMA-3.1-8B-INSTRUCT.

Method	ACC	Corr.	Cohe.
A-MEM	0.78	0.795	0.81
Ours	0.88	0.92	0.92

TABLE II
F1 SCORE COMPARISON ON PERLTQA USING LLAMA-3.1-8B-INSTRUCT.

Method	Profile	Social Relationship	Dialogues	Event	Overall
A-MEM	28.3	18.2	11.7	10.4	12.4
Ours	48.3	47.2	29.5	46.0	40.1

- **Temporal neighbors:** memories occurring near the anchor in the conversation sequence, preserving local context and episodic continuity.

The two neighborhoods are combined into a shared candidate pool, allowing retrieval to capture both distant semantic associations and nearby conversational evidence.

C. Path-Aware Candidate Selection

Expanded memories are evaluated in the context of both the query and the anchor from which they were discovered. A cross-encoder therefore estimates whether a candidate provides useful complementary information along the current retrieval path, rather than judging each memory independently.

The highest-ranked candidates are retained as evidence and simultaneously become new anchors for the next round of exploration. This process continues for multiple hops, progressively extending the retrieved evidence while filtering noisy or redundant memories.

Finally, all selected memories are aggregated and provided to the LLM as the context for answer generation.

III. EXPERIMENT

We evaluate our method on two long-term memory benchmarks, GVD and PerLTQA. GVD contains multi-turn dialogues between virtual users and an assistant, while PerLTQA focuses on personalized long-term memory over extended contexts. Following MemoryOS [5], we report Memory Retrieval Accuracy (Acc.), Response Correctness (Corr.), and Contextual Coherence (Cohe.) on GVD, with GPT-4o-mini used for evaluation.

As shown in Table I and Table II, our method consistently outperforms **A-MEM**. On GVD, accuracy improves from 0.78 to 0.88, together with higher correctness and coherence. On PerLTQA, the overall F1 increases from 12.4 to 40.1, with particularly large gains on Social Relationship and Event questions. These results demonstrate the effectiveness of our retrieval strategy across different long-term memory benchmarks.

ACKNOWLEDGMENT

The entire structure of the paper, including the initial and subsequent versions, was created by the authors. LLM tools were used for language editing, including formatting, grammar correction in the Abstract, Introduction, Methodology, and Experiment sections. All ideas, analyses, and conclusions are solely those of the authors.

REFERENCES

[1] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, "Voyager: An open-ended embodied agent with large language models," *arXiv preprint arXiv:2305.16291*, 2023.

[2] W. Xu, Z. Liang, K. Mei, H. Gao, J. Tan, and Y. Zhang, "A-mem: Agentic memory for llm agents," *Advances in Neural Information Processing Systems*, vol. 38, pp. 17 577–17 604, 2026.

[3] C. Packer, S. Wooders, K. Lin, V. Fang, S. G. Patil, I. Stoica, and J. E. Gonzalez, "Memgpt: Towards llms as operating systems," *arXiv preprint arXiv:2310.08560*, 2023.

[4] A. Maharana, D.-H. Lee, S. Tulyakov, M. Bansal, F. Barbieri, and Y. Fang, "Evaluating very long-term conversational memory of llm agents," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024, pp. 13 851–13 870.

[5] J. Kang, M. Ji, Z. Zhao, and T. Bai, "Memory os of ai agent," in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025, pp. 25 972–25 981.