MEMORYBENCH: A DIAGNOSTIC BENCHMARK FOR PERSISTENT MEMORY IN LLM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Persistent memory is essential for long-lived LLM agents, yet existing benchmarks primarily evaluate retrieval from available context rather than memory curation: the ability to decide what to store, update, delete, and consolidate over time. We introduce MEMORYBENCH, a diagnostic benchmark for evaluating end-to-end persistent memory systems when prior sessions are no longer accessible.[1]

MEMORYBENCH comprises 71 tests across 7 failure categories spanning 20 multi-session user profiles. Each category corresponds to a distinct memory operation—handling contradictions, expiring stale information, filtering noise, inferring implicit preferences, consolidating related facts, and synthesizing across sessions—enabling fine-grained diagnosis of memory system behavior. The benchmark includes structured ground-truth annotations and a controlled evaluation protocol that standardizes the underlying LLM, embeddings, and conversations to isolate memory pipeline differences.

We demonstrate that MEMORYBENCH reveals complementary strengths and blind spots across representative memory architectures, and we release it as an open, extensible framework to support systematic evaluation and development of persistent memory systems for LLM agents.

1 INTRODUCTION

Persistent memory is essential for long-lived LLM agents. Personal assistants, coding copilots, research collaborators, and enterprise agents must remember information across sessions to provide coherent and personalized interaction. Over time, users change roles, relocate, revise plans, and express evolving preferences. Memory systems must therefore decide not only what to remember, but also what to update, delete, consolidate, or ignore as conversations unfold.

Two dominant paradigms have emerged. In **agent-managed memory**, the conversational LLM directly controls memory operations such as adding, updating, or deleting entries (Park et al., 2023; Packer et al., 2023). In **external memory services**, memory management is handled by automated extraction pipelines that operate alongside the conversational model. Both paradigms are actively deployed, yet empirical comparison under controlled conditions remains limited.

Existing benchmarks primarily evaluate memory *retrieval*—whether models can access relevant information from long contexts (Liu et al., 2024; Bai et al., 2024) or multi-session transcripts (Maharana et al., 2024). These settings assume that past information remains available at inference time. In contrast, deployed memory systems operate under context loss: prior sessions are not directly accessible, and behavior depends entirely on what the memory pipeline previously stored and how effectively it can retrieve it. Evaluating such systems requires moving beyond retrieval accuracy toward structured analysis of persistent memory behavior.

We introduce MEMORYBENCH, a diagnostic benchmark for end-to-end persistent memory systems. The benchmark comprises 71 tests across 7 failure categories spanning 20 multi-session user profiles. Each category corresponds to a distinct memory operation—handling contradictions, expiring stale information, filtering noise, inferring implicit preferences, consolidating related facts, and

[1] Code and benchmark available at: `https://anonymous.4open.science/r/Memory-Benchmark-1-1006/`

Table 1: MEMORYBENCH diagnostic categories (71 tests across 20 profiles).

Category	N	Example
Contradiction Update	18	"Where am I based now?" (NYC → SF)
Temporal Relevance	10	"Am I still on-call this week?"
Noise Resistance	9	"Do you remember the coffee incident?"
Implicit Preference	10	"How should you explain things to me?"
Simple Recall	8	"What framework am I using?"
Consolidation	8	"How has my model's performance evolved?"
Cross-Session Synthesis	8	"What's my overall learning journey?"
Total	**71**	

synthesizing across sessions. By organizing evaluation around these categories, MEMORYBENCH reveals not only overall accuracy but also *why* systems fail.

Importantly, MEMORYBENCH evaluates complete memory pipelines—including storage, indexing, and retrieval—under controlled conditions. We standardize the underlying LLM, embeddings, and conversations to isolate differences in memory management strategies. This design enables principled comparison between agent-managed and external memory paradigms while reflecting the operational realities of deployed systems.

Contributions. (1) We introduce MEMORYBENCH, a structured diagnostic benchmark for persistent memory systems with multi-session profiles and category-level evaluation. (2) We provide a controlled comparison of representative agent-managed and external memory approaches, revealing complementary strengths across memory operations. (3) We release MEMORYBENCH as an extensible framework to support systematic evaluation and development of memory systems for long-lived LLM agents.

2 MEMORYBENCH

MEMORYBENCH is designed around three principles.

(1) Multi-session realism. Persistent memory failures arise over time rather than within a single conversation. Each profile spans five sessions reflecting natural evolution (e.g., job transitions, relocations, shifting priorities, evolving preferences).

(2) Structured ground truth. Each profile includes explicit annotations specifying what information should be stored, what supersedes earlier facts, and what constitutes irrelevant noise. Session 5 contains evaluation queries aligned with these annotations.

(3) Diagnostic categorization. Tests are grouped into seven failure categories corresponding to distinct memory operations, enabling fine-grained analysis of system behavior beyond aggregate accuracy.

Profiles span ten professional domains (e.g., software engineering, data science, game development, clinical research). Each contains five multi-turn sessions averaging 3–5 turns. Sessions 1–4 embed memory challenges within natural dialogue. Session 5 contains evaluation queries assessed against gold references.

2.1 EVALUATION PROTOCOL

For each profile, we initialize an empty memory store and sequentially process Sessions 1–4 using the memory system under study. At evaluation time (Session 5), prior conversations are not available; systems must rely exclusively on persisted memory. For each query, the system retrieves the top-5 memories via semantic similarity and generates a response.

All systems use the same conversational model (GPT-4o-mini), embedding model (text-embedding-3-small), and conversation transcripts to isolate differences in memory management strategies. Responses are scored as $\{$correct, partially correct, incorrect$\}$.

Table 2: Overall accuracy across 71 memory tests (20 profiles).

System	Strict	w/ Partial	Correct	Partial	Incorrect
Agent-Driven	62.0%	**75.4%**	44	19	8
LangMem	62.0%	71.1%	44	13	14
Mem0	45.1%	59.9%	32	21	17

2.2 Judge Reliability

To assess scoring reliability, two annotators independently reviewed all 71 evaluation cases (query, system answer, gold reference) and assigned labels using the same three-way scheme. Agreement with the automated judge was 97.2% (69/71). The two disagreements involved borderline distinctions between partially correct and correct responses and were resolved through discussion. This analysis suggests that scoring noise does not materially affect aggregate comparisons.

3 Systems Under Study

We evaluate three representative persistent memory systems under controlled conditions.

Agent-Driven. A conversational design in which a single LLM call per turn produces both the user response and memory operations (`add`, `update`, `delete`). Memory decisions are generated directly from conversational context. When the store exceeds 20 entries, a consolidation step merges related memories.

Mem0. An LLM-based extraction pipeline that processes completed conversations to extract candidate memories, followed by deduplication against existing entries.

LangMem. A semantic memory manager that tracks memory objects across invocations, providing existing memories during subsequent extraction to support updates when contradictions arise.

A no-memory baseline (current session only) achieves 11% strict accuracy, confirming that persistent memory materially improves performance.

All systems employ top-5 semantic retrieval at evaluation time. In deployed agents, utility depends on both correct storage and effective retrievability; MemoryBench therefore evaluates complete memory pipelines rather than isolated storage decisions.

4 Results

Our goal is not to establish a leaderboard or declare one memory paradigm superior. Rather, we use representative systems to demonstrate how MemoryBench's taxonomy reveals structured differences in persistent memory behavior. Results are therefore illustrative rather than definitive. Absolute performance may vary with different prompts, embedding models, retrieval configurations, or implementation details. Our controlled setup (same LLM, embeddings, and conversations) aims to isolate memory management strategies, but the primary contribution of this work is the diagnostic framework itself.

Agent-Driven and LangMem achieve identical strict accuracy (62.0%), while Mem0 performs lower (45.1%). Although aggregate metrics appear similar between the top two systems, category-level analysis reveals meaningful differences (Figure 1).

Contradiction update and temporal relevance. Agent-Driven performs best on contradiction updates (83.3%) and temporal relevance (80.0%), outperforming both external systems. These categories require recognizing superseded or expired information during conversation.

Cross-session synthesis. LangMem achieves the highest performance on cross-session synthesis (50.0% vs. 12.5% for Agent-Driven). These tests require integrating information distributed across multiple sessions. Error analysis suggests that synthesizing multi-aspect information remains challenging for all systems.

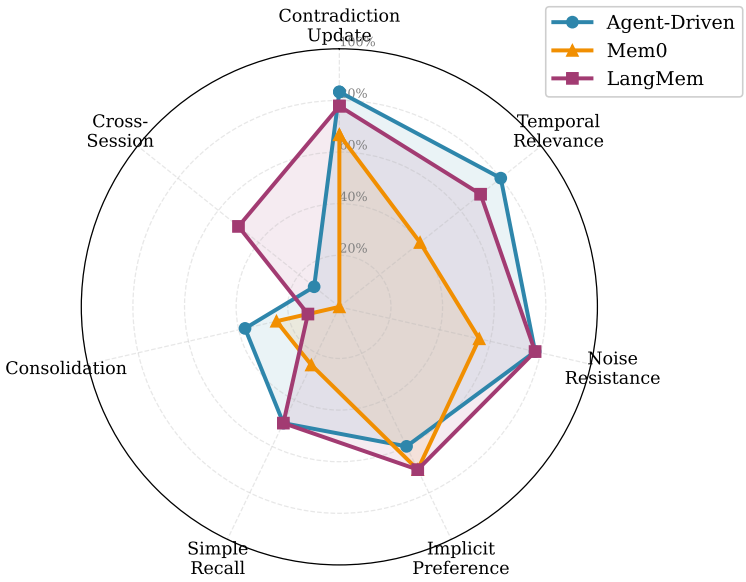

Figure 1: Per-category strict accuracy across three systems. Performance profiles differ substantially across memory operations.

Implicit preferences and noise resistance. External systems match or exceed Agent-Driven on implicit preference extraction (70.0% vs. 60.0%). Noise resistance results are comparable between Agent-Driven and LangMem (77.8%), with Mem0 performing lower.

Prompt sensitivity. Improving the Agent-Driven extraction prompt increases strict accuracy from 46.5% to 62.0% (+15.5pp), without architectural changes. This suggests that memory extraction quality substantially influences persistent memory performance.

5 Discussion

MemoryBench demonstrates that persistent memory systems exhibit complementary strengths across memory operations. Agent-managed systems benefit from conversational context when handling temporal updates and contradictions, while structured extraction pipelines can better aggregate information across sessions.

Operational trade-offs. Agent-Driven maintains a smaller memory store (8.8 entries per profile) compared to Mem0 (23.0 entries), which may reduce retrieval noise but risks missing long-range synthesis opportunities. External systems leverage full-store access during extraction but may retain redundant or outdated entries if updates are imperfect.

Extraction as a bottleneck. The substantial improvement from prompt refinement suggests that extraction quality, rather than paradigm choice alone, is a central determinant of memory performance.

Limitations. The benchmark includes 71 tests and serves as an initial diagnostic suite rather than a comprehensive evaluation standard. Profiles are synthetic and may not capture the full diversity of real-world interactions. We evaluate a single LLM and embedding model, and results may vary across model families. Experiments were conducted as single trials due to cost constraints.

Retrieval–curation coupling. MemoryBench evaluates end-to-end persistent memory systems. Storage correctness and retrievability are operationally intertwined: a memory that cannot be retrieved is functionally equivalent to a missing memory. While categories align with specific

memory operations, results reflect complete pipeline behavior. Future work may further disentangle storage and retrieval via oracle evaluation or explicit recall metrics.

Benchmark purpose. The system comparison presented here is intended to illustrate the diagnostic value of MEMORYBENCH, not to provide a definitive ranking of memory architectures. Performance is sensitive to configuration choices, including prompt design, retrieval parameters, embedding models, and implementation details. Different configurations could shift aggregate scores. The central contribution of this work is the taxonomy and evaluation protocol, which enable structured analysis of such differences.

6 CONCLUSION

We presented MEMORYBENCH, a diagnostic benchmark for evaluating persistent memory systems in long-lived LLM agents. By organizing evaluation around seven memory operations and comparing representative paradigms under controlled conditions, the benchmark reveals structured differences that aggregate accuracy obscures. MEMORYBENCH provides an extensible framework for analyzing and improving memory system behavior in deployed LLM agents.

REFERENCES

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. LongBench: A bilingual, multitask benchmark for long context understanding. *arXiv preprint arXiv:2308.14508*, 2024.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 2024.

Adyasha Maharana, Dong-Ho Lee, Sergey Tuber, Tristan Olausson, and Mohit Bansal. LoCoMo: Long-context conversational memory benchmark. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

Charles Packer, Vivian Fang, Shishir G Patil, Kevin Lin, Sarah Wooders, and Joseph E Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Joon Sung Park, Joseph C O'Brien, Carrie J Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023.