Anatomy of Agentic Memory: Taxonomy and Empirical Analysis of Evaluation and System Limitations

Dongming Jiang[α], Yi Li[α], Songtao Wei[α], Jinxin Yang[α], Ayushi Kishore[β], Alysa Zhao[γ], Dingyi Kang[α],

Xu Hu[α], Feng Chen[α], Qiannan Li[β] and Bingzhe Li[α,*]

[α]University of Texas at Dallas [β]University of California, Davis [γ]Texas A&M University

{dongming.jiang, yi.li3, songtao.wei, jinxin.yang, dingyi.kang, xu.hu, feng.chen, bingzhe.li}@utdallas.edu

{aykishore, qnli}@ucdavis.edu; alysazhao111@tamu.edu

[*]Corresponding author GitHub Repo

Abstract

Agentic memory systems enable large language model (LLM) agents to maintain state across long interactions, supporting long-horizon reasoning and personalization beyond fixed context windows. Despite rapid architectural development, the empirical foundations of these systems remain fragile: existing benchmarks are often underscaled, evaluation metrics are misaligned with semantic utility, performance varies significantly across backbone models, and system-level costs are frequently overlooked. This survey presents a structured analysis of agentic memory from both architectural and system perspectives. We first introduce a concise taxonomy of Memory-Augmented Generation (MAG) systems based on four memory structures. Then, we analyze key pain points limiting current systems, including benchmark saturation effects, metric validity and judge sensitivity, backbone-dependent accuracy, and the latency and throughput overhead introduced by memory maintenance. By connecting the memory structure to empirical limitations, this survey clarifies why current agentic memory systems often underperform their theoretical promise and outlines directions for more reliable evaluation and scalable system design. To facilitate future research, we maintain an open-source repository of surveyed papers, benchmarks, and resources.[1]

1 Introduction

Large language model (LLM) agents are increasingly expected to operate over long time horizons, maintaining user preferences, accumulating task-relevant knowledge, etc. (Brown et al., 2020; Achiam et al., 2023; Wei et al., 2022). However, fixed context windows fundamentally limit their ability to retain and manipulate persistent state (Brown et al., 2020; Beltagy et al., 2020; Liu et al., 2024; Press et al., 2021). To address this constraint, Memory-Augmented Generation (MAG) extends memory beyond the prompt, enabling agents to store, retrieve, and update information across interactions (Xu et al., 2025c; Nan et al., 2025; Chhikara et al., 2025; Jiang et al., 2026a; Liu et al., 2026). While this paradigm has rapidly evolved from lightweight semantic stores to entity-centric, reflective, and hierarchical designs, empirical understanding remains limited: reported gains are inconsistent across benchmarks, highly backbone-dependent, and lack principled guidance on evaluation and system-level cost.

These challenges stem in part from underspecified design trade-offs and inadequate evaluation. Benchmarks are often underscaled relative to modern context windows, metrics emphasize surface overlap over semantic utility, and system-level costs such as latency and throughput degradation are rarely measured. As a result, complex memory systems are frequently tested in settings where simpler full-context or retrieval baselines suffice, obscuring their true benefits and limitations.

In this paper, we provide a structured analysis of agentic memory systems from both architectural and empirical perspectives. 1) We introduce a concise taxonomy of Memory-Augmented Generation organized around four memory structures: Lightweight Semantic, Entity-Centric and Personalized, Episodic and Reflective, and Structured and Hierarchical. Defined by how memory is organized and manipulated, this taxonomy establishes a principled basis for analyzing system behavior. 2) Building on this framework, we identify key bottlenecks limiting reliability and scalability, including benchmark saturation, metric misalignment (e.g., F1 vs. semantic correctness), prompt sensitivity in LLM-as-a-judge evaluation, backbone dependence, and system-level costs such as retrieval latency,

[1] https://github.com/FredJiang0324/Anatomy-of-Agentic-Memory. Given the rapid evolution of research in agentic memory, we may have inadvertently overlooked some relevant studies. We welcome you to suggest missing papers by emailing us or opening an issue on our GitHub repository.

Table 1: Comparison with related surveys on memory for LLM-based agents. ✓ indicates the topic is systematically discussed; (✓) indicates partial or brief coverage; × indicates the topic is not addressed.

Survey	Taxonomy Focus	Memory Mgmt. & Policy	Benchmark Saturation	Metric Validity	Backbone Sensitivity	System Cost & Latency
The AI Hippocampus (Jia et al., 2026)	Brain-inspired: implicit, explicit, agentic	(✓)	×	×	×	×
Memory in the Age of AI Agents (Hu et al., 2025)	Forms–functions–dynamics	(✓)	×	×	×	×
Toward Efficient Agents (Yang et al., 2026c)	Efficiency-focused: memory, tool learning, planning	✓	×	×	×	✓
Rethinking Memory Mechanisms (Huang et al., 2026)	Substrate–cognition–subject	✓	(✓)	×	×	×
From Storage to Experience (Luo et al., 2026)	Evolutionary: storage–reflection–experience	(✓)	×	×	×	×
Graph-based Agent Memory (Yang et al., 2026a)	Graph-oriented lifecycle	✓	(✓)	×	×	×
Taxonomy and Empirical Analysis (Ours)	Structural + Empirical analysis	✓	✓	✓	✓	✓

update overhead, and throughput degradation.

By linking memory structures to empirical limitations, this survey clarifies why current agentic memory systems often fall short of their theoretical promise. Rather than advocating a single "best" design, we provide a diagnostic framework to explain when specific memory structures are effective, when they fail, and what trade-offs they entail. Our analysis offers guidance for designing more robust benchmarks, more reliable evaluation protocols, and more scalable agentic memory systems.

Difference from other surveys: While existing surveys (Jia et al., 2026; Hu et al., 2025; Yang et al., 2026c,a; Luo et al., 2026; Huang et al., 2026) primarily operate at the *theoretical* level by cataloguing architectures, defining conceptual taxonomies, and drawing cognitive science analogies, our survey bridges the gap *from theory to practice*. Our taxonomy is structure-oriented, not only discussing various memory structure designs, but also highlighting the memory management frameworks and optimization strategies. In addition, we provide comprehensive evaluations across multiple benchmarks. Specifically, we conduct systematic analyses of agentic memory systems on benchmark saturation, metric validity, backbone sensitivity, and maintenance overhead, overlooked in prior surveys yet critical for understanding why current MAG systems often fall short of their theoretical promise. A detailed comparison is presented in Table 1.

2 Background

Agentic memory extends retrieval-based generation by introducing persistent, writable memory that evolves across interactions, enabling an LLM agent to store, update, and reuse information over time. Formally, at step t, the agent conditions on observations o_t and an external memory state $\mathcal{M}_t$:

$$y_t \sim f_\theta\Big(\phi(o_t, s_t) \;\oplus\; \psi(\mathcal{M}_t; q_t)\Big), \qquad (1)$$

where y_t denotes the output, s_t additional agent state, and $\psi(\mathcal{M}_t; q_t)$ retrieves memory given query q_t. The operator $\oplus$ represents integration (e.g., prompt concatenation or structured fusion). Crucially, memory affects behavior through the explicit retrieval term $\psi(\mathcal{M}_t; q_t)$ rather than updates to θ.

Two coupled processes are operated in agentic memory: inference-time recall and memory update. At each step, the agent retrieves relevant information from an external memory store to condition its decision, and subsequently writes, updates, or consolidates memory to maintain a useful long-term state. Unlike parametric learning, this mechanism influences behavior through explicit read–write operations over an evolving memory state rather than by modifying model weights. A formalization of these operations including query generation, utility-aware retrieval, and memory actions such as store, summarize, link, and delete is provided in Appendix B.

3 Taxonomy of Agentic Memory

We introduce a concise taxonomy of Memory-Augmented Generation organized around four memory structures: Lightweight Semantic, Entity-Centric and Personalized, Episodic and Reflective, and Structured and Hierarchical. Each category is further split into subcategories as shown in Figure 1 in Appendix A.

3.1 Lightweight Semantic Memory

Lightweight Semantic Memory is the simplest and most widely used form of MAG, where memory consists of independent textual units embedded in a vector space and retrieved via top-k similarity

search. Entries are typically append-only or minimally filtered, with no explicit structural relations between them.

RL-Optimized Semantic Compression: These schemes treat memory as a fixed-size semantic store and apply RL to optimize how information is compressed, retained, or overwritten under context constraints (Wang et al., 2025b; Yan et al., 2025; Yuan et al., 2025). Memory remains largely unstructured and textual, with learning focused on efficient content selection. For example, MemAgent (Yu et al., 2025) trains a latent token-level memory using multi-conversation RL to manage ultra-long contexts, while MemSearcher (Yuan et al., 2025) formulates multi-turn search and memory updates as an end-to-end RL problem, iteratively compressing semantic memory to enable scalable multi-hop reasoning without relying on full dialogue history.

Heuristic / Prompt-Optimized: These approaches manage memory through prompt design or heuristic rewriting with a flat, compressed textual summary of prior steps generated via engineered instructions, reducing context length but remaining unstructured. Similar prompt-driven compression strategies appear in prior work (Zhao et al., 2025; Wu and Li, 2025; Liu et al., 2026; Li et al., 2026b). For example, ACON (Kang et al., 2025c) learns natural-language compression guidelines to selectively summarize long interaction histories, reducing context by up to 54% without RL or fine-tuning, while CISM (Liu et al., 2025c) condenses each reasoning and action step into compact semantic representations to enable long-horizon execution under context constraints without explicit external memory retrieval.

Context Window Management: This category manages the model's working context within a single task, without accumulating memory across sessions. Prior interactions are folded, summarized, or reorganized to fit within a bounded window, prioritizing local reasoning efficiency over long-term storage or reuse (Zhu et al., 2025; Sakib et al., 2025). For example, AgentFold (Ye et al., 2025) treats context as a dynamic workspace and learns multi-scale folding operations to condense long trajectories, while Context-Folding Agent (Sun et al., 2025b) trains an RL-based policy that branches sub-tasks and compresses completed segments.

Token-Level Semantic Memory: This category encodes memory at the token level using dedicated memory tokens or compressed latent panels. These representations primarily capture semantic content, aiming to improve long-context handling with minimal overhead (Wu et al., 2025b; Zhang et al., 2025c; Yang et al., 2024b). Memory entries are independent and inexpensive to store or retrieve, making them suitable for short- to medium-horizon recall, but limited for precise state tracking or long-term reasoning. For example, MemGen (Zhang et al., 2025c) augments a frozen LLM with on-demand latent token memory via an RL-trained trigger and LoRA-based weaving, while TokMem (Wu et al., 2025b) replaces lengthy procedural prompts with trainable memory tokens to enable constant-size context management and scalable skill reuse.

3.2 Entity-Centric and Personalized Memory

Entity-centric and personalized memory organizes information around explicit entities such as users, tasks, or preferences, using structured records or attribute–value pairs. A predefined schema governs how information is stored, updated, and retrieved.

Entity-Centric Memory: Entity-centric memory organizes information around explicit entities and their attributes, maintaining structured, persistent records rather than raw dialogue (Modarressi et al., 2023; Liu et al., 2021, 2025a). For example, A-MEM (Xu et al., 2025c) builds interconnected knowledge notes with structured attributes and LLM-generated links; Memory-R1 (Yan et al., 2025) formulates entity memory management as an RL problem over a persistent entity–fact bank.

Personalized Memory: Personalized memory maintains persistent user profiles that integrate short- and long-term preferences to support adaptive, identity-consistent behavior across sessions (Zhong et al., 2024; Li et al., 2025a; Kwon et al., 2025; Liu et al., 2025a; Mao et al., 2026; Su et al., 2026). For example, PAMU (Sun et al., 2025a) combines sliding windows with moving averages to track evolving preferences, EgoMem (Yao et al., 2025) constructs lifelong multimodal profiles with conflict-aware updates, and MemOrb (Huang et al., 2025a) stores compact reflective memories for continual improvement.

3.3 Episodic and Reflective Memory

Episodic and reflective memory adds temporal abstraction by organizing interactions into episodes or higher-level summaries. These systems periodically consolidate experience through summarization or reflection, producing compact representations of salient events over time.

Episodic Buffer w/ Learned Control: memory in these work consists of episodic interaction records maintained in a bounded buffer and dynamically inserted, retained, or deleted through learned policies (Du et al., 2025a; Zhang et al., 2025f; Icarte et al., 2020). For example, MemR^3 (Du et al., 2025a) models retrieval as a closed-loop retrieve–reflect–answer process; and the Act of Remembering (Icarte et al., 2020) formulates remembering as a control problem in POMDPs with a fixed-capacity episodic buffer.

Episodic Recall for Exploration: These methods leverage episodic memory to improve exploration and credit assignment in partially observable or long-horizon settings. Past experiences are stored and selectively retrieved to guide decision-making (Na et al., 2024; Adamyan et al., 2025). For example, EMU (Na et al., 2024) maintains large-capacity episodic memories indexed by learned embeddings to accelerate cooperative MARL exploration, while SAM2RL (Adamyan et al., 2025) uses a visual memory bank as an episodic buffer and trains a PPO policy to manage memory replacement, outperforming heuristic updates under challenging conditions.

Episodic Reflection & Consolidation: This subcategory reflects and consolidates episodic experiences into compact representations (Tan et al., 2025b; Kim et al., 2025; Ouyang et al., 2025; Dong et al., 2025; Lee et al., 2024). The objective is to balance memory capacity with long-term reasoning utility. For example, MemP (Fang et al., 2025b) distills trajectories into procedural abstractions for continual refinement and transfer; LEGOMem (Han et al., 2025) constructs modular, role-aware procedural memories for multi-agent coordination; and TiMem (Li et al., 2026a) introduces a temporal-hierarchical memory tree for structured consolidation and scalable long-horizon personalization without RL or fine-tuning.

Episodic Utility Learning: Episodic memories in these setting are augmented with learned value or utility signals that evolve over time, enabling selective retention and retrieval based on both semantic relevance and estimated long-term usefulness (Zhou et al., 2025a; Cao et al., 2025). For example, MemRL (Zhang et al., 2026c) associates utility Q-values with intent–experience pairs and updates them online to balance stability and plasticity without fine-tuning, while Memory-T1 (Du et al., 2025b) learns a temporal-aware retrieval policy via GRPO to optimize accuracy, grounding, and chronological consistency in long-context dialogue.

3.4 Structured and Hierarchical Memory

Structured and hierarchical memory systems impose explicit organization over stored information. Hierarchical designs partition memory into multiple tiers (e.g., short- and long-term stores), while structured approaches encode relationships among memory elements using graphs or other formal relational representations.

Graph-Structured Memory: Graph-structured memory represents information as nodes and edges capturing semantic, temporal, causal, or entity-level relations, enabling reasoning over structured subgraphs (Zhang et al., 2025b,d; Jiang et al., 2026c; Tao et al., 2026; Zhang et al., 2026d; Hu et al., 2026b). This design supports multi-hop inference, provenance tracking, and coherent long-horizon reasoning. For example, MAGMA (Jiang et al., 2026a) organizes memory across semantic, temporal, causal, and entity graphs; Zep (Rasmussen et al., 2025) constructs a bi-temporal knowledge graph with episodic and semantic layers; SGMem (Wu et al., 2025a) models dialogue as sentence-level graphs; and LatentGraphMem (Zhang et al., 2026d) integrates latent graph encoding with a compact symbolic subgraph to balance stability, efficiency, and interpretability.

OS-Inspired & Hierarchical Memory: OS-inspired and hierarchical memory systems organize information into multi-tier storage layers (e.g., short-term, episodic, long-term), dynamically moving and consolidating data to balance scalability, retention, and adaptive forgetting (Xu, 2025; Ouyang, 2025; Zhang et al., 2025e; Jia et al., 2025; Li et al., 2026a). For example, MemGPT (Packer et al., 2023) enables LLM-driven memory paging across tiers; MemoryOS (Kang et al., 2025a) implements a modular three-level hierarchy; EverMemOS (Hu et al., 2026a) and HiMem (Zhang et al., 2026b) consolidate episodic and semantic traces for long-horizon adaptation; and MeMAD (Ling et al., 2025) stores structured debate experiences for reusable reasoning.

Policy-Optimized Memory Management: Policy-optimized memory management treats storage, update, consolidation, and deletion as learnable decisions, using reinforcement learning or hybrid training to optimize long-horizon rewards (Liu et al., 2025b; Xu et al., 2025b; Kang et al., 2025b; Du et al., 2025b). For example, MEM1 (Zhou

et al., 2025b) learns to maintain a compact internal state with constant-memory operations; and Mem-α (Wang et al., 2025b) trains an RL policy to manage multi-component external memory under ultra-long contexts; and AtomMem (Huo et al., 2026) decomposes memory into CRUD actions to learn task-aligned control strategies. While enabling adaptive and scalable management, these approaches introduce greater system complexity and nontrivial maintenance overhead.

3.5 Discussion

The four categories described above capture the dominant memory structures used in contemporary MAG systems. While individual systems may combine multiple mechanisms, each can typically be characterized by a primary memory organization that governs its behavior. This structure-first taxonomy provides a foundation for understanding how design choices in agentic memory influence accuracy, efficiency, and reliability. In the next section, we build on this taxonomy to analyze the empirical limitations and pain points that arise across current MAG systems.

4 Evaluation and Pain Points

In this section, we move beyond taxonomy to empirically analyze the practical bottlenecks hindering robust deployment. While theoretical architectures are promising, real world utility is strictly constrained by evaluation validity, system efficiency, and backbone reliability. We dissect these challenges across four critical dimensions:

1. Benchmark Validity: Are we testing memory or just context length?
2. Metric Reliability: Can lexical metrics capture semantic coherence?
3. System Efficiency: The "Agency Tax" of latency and cost.
4. Backbone Sensitivity: The "Silent Failure" of memory operations in open-weight models.

4.1 Experimental Setup

We evaluate representative MAG systems spanning the four taxonomy categories introduced in Section 3. Six memory architectures are selected: AMem (Xu et al., 2025c), MemoryOS (Kang et al., 2025a), Nemori (Nan et al., 2025), MAGMA (Jiang et al., 2026a), SimpleMEM (Liu et al., 2026) and MemSkill (Zhang et al., 2026a) as shown in Table 8 of Appendix E. All systems are configured to follow their default or recommended settings, except where modifications are required to ensure comparability. We employ a suite of Large Language Models (LLMs) to serve as the agent controller including `gpt-4o-mini` (Achiam et al., 2023) and `Qwen-2.5-3B` (Yang et al., 2024a).

4.2 Benchmark Scalability: The Context Saturation Risk

A key motivation for agentic memory is to support reasoning beyond a model's finite context window. Yet as LLM windows expand (e.g., 128k to 1M tokens), many benchmarks risk *context saturation*, where all relevant information fits within a single prompt, making external memory seemingly unnecessary. In this section, rather than comparing performance, we examine the intrinsic properties of existing datasets to evaluate their continued validity in the long-context era.

4.2.1 Dimensions of Limitation

First, we evaluate benchmark scalability along three structural axes: volume, interaction depth, and entity diversity, to assess their saturation risk under long-context LLMs as shown in Table 2.

Volume (Total Token Load). This dimension captures the aggregate information size a model must process. Benchmarks such as *HotpotQA* ($\sim$1k tokens) and *MemBench* ($\sim$100k tokens) fall within a 128k context window, implying high theoretical saturation risk. *LoCoMo* ($\sim$20k tokens) similarly remains comfortably in-window for modern models. Only datasets that substantially exceed the active window (e.g., *LongMemEval-M* at >1M tokens) structurally require external memory.

Interaction Depth (Temporal Structure). Beyond raw volume, scalability depends on how information unfolds across sessions. Single-turn QA (e.g., *HotpotQA*) imposes minimal temporal dependency, whereas multi-session settings (e.g., *LoCoMo* with 35 sessions) introduce longitudinal reasoning. However, unless cross-session dependencies exceed the active window or require persistent state tracking beyond prompt capacity, such datasets may still be solvable through direct in-context aggregation rather than true memory management.

Entity Diversity (Relational Complexity). This axis measures how many distinct entities or conceptual threads must be tracked simultaneously. Low-diversity benchmarks permit near-isolated re-

Table 2: Structural saturation risk of memory benchmarks. Benchmarks are analyzed based on intrinsic statistics rather than model performance. *Saturation Risk* is a heuristic estimate of whether a long-context LLM may solve the benchmark through direct prompting.

Benchmark	Scalability Dimensions			Theoretical Saturation Risk
	Avg. Volume	Interaction Depth	Entity Diversity	
HotpotQA (Yang et al., 2018)	∼1k Tokens	Single Turn	Low	High (Trivial for Context Window)
LoCoMo (Maharana et al., 2024)	∼20k Tokens	35 Sessions	High	Moderate (Requires Reasoning)
LongMemEval-S (Wu et al., 2024)	103k Tokens	5 Core Abilities	High	Moderate (Borderline)
LongMemEval-M (Wu et al., 2024)	>1M Tokens	5 Core Abilities	High	Low (Requires External Memory)
MemBench (Tan et al., 2025a)	∼100k Tokens	Fact/Reflection	Medium	High (Fits in 128k Window)

trieval, while higher-diversity settings (e.g., *LoCoMo*, *LongMemEval*) increase interference and relational reasoning demands. Nevertheless, if entity interactions remain bounded within context limits, structured external memory may not be strictly necessary.

Discussion: Taken together, these dimensions show that saturation risk is determined not by surface difficulty but by whether a benchmark's structural properties exceed the representational capacity of long-context LLMs.

4.2.2 Context Saturation Gap as an Empirical Diagnostic

To address these limitations, we propose that future evaluations explicitly quantify the *Context Saturation Gap* (Δ), defined as the performance difference between a Memory-Augmented Agent (MAG) and a brute-force Full-Context baseline under the same backbone and evaluation protocol:

$$\Delta = \text{Score}_{\text{MAG}} - \text{Score}_{\text{FullContext}} \quad (2)$$

A large positive Δ indicates that external memory provides an advantage beyond simply placing all available evidence in the prompt, especially in out-of-window or lost-in-the-middle regimes. However, Δ should be viewed as a diagnostic rather than a strict pass/fail criterion, since benchmarks may still evaluate memory through efficiency, updateability, robustness, or evidence faithfulness even when Full-Context remains competitive.

Table 2 therefore reports structural saturation risk rather than an empirical saturation test. Datasets with limited volume and shallow complexity should be paired with Full-Context baselines before drawing strong conclusions about memory-specific benefits.

4.3 LLM-as-a-Judge Evaluation

Traditional lexical metrics (e.g., F1, BLEU) emphasize surface-level token overlap, which is insufficient for agentic memory tasks where the goal is accurate retrieval and coherent synthesis rather than exact phrasing. To better capture semantic correctness, we adopt an LLM-based evaluator (gpt-4o-mini) as a proxy for human judgment. In this section, we assess the reliability of this protocol by analyzing the misalignment between lexical and semantic metrics and demonstrating the stability of our system rankings across competitive evaluation settings.

4.3.1 The Misalignment Gap

Do lexical metrics correctly identify the best memory system? To examine this, we compared system rankings produced by F1-score with those generated by an LLM-based judge across six representative architectures on the *LoCoMo* dataset.

Table 3 (Left) reveals a significant disconnect. Lexical metrics often fail to capture the strengths of abstractive memory systems. For example, AMem achieves solid semantic performance (Rank 4 across prompts) due to its logical coherence, yet it is heavily penalized by F1 (Rank 5, Score 0.116) because it does not rely on verbatim overlap. In contrast, SimpleMem receives a relatively higher F1 score (0.268) despite demonstrating limited ability to synthesize complex answers (semantic score < 0.30). This divergence indicates that optimizing solely for F1 may favor surface-level memorization over genuine reasoning and memory integration.

4.3.2 Semantic Judge Robustness Across Prompts

A common concern with LLM-as-a-judge is "prompt overfitting," where a system appears strong only under a specific grading instruction. To ensure fairness and generality, we evaluated all architectures using three distinct prompt protocols derived from different sources (details in Appendix D.3).

As shown in Table 3 (Right), compared with F1-based rankings, the semantic judge exhibits strong robustness: the relative ordering of architectures remains highly consistent across different rubrics.

Table 3: Robustness of system ranking across evaluation protocols. We compare Lexical metrics (F1) against LLM-based semantic evaluation using three distinct prompt sources: MAGMA, Nemori, and SimpleMem.

Method	Lexical Metric		Semantic Judge Score (Rank)		
	F1-Score	Rank	Prompt 1 (MAGMA)	Prompt 2 (Nemori)	Prompt 3 (SimpleMem)
AMem (Xu et al., 2025c)	0.116	5	0.480 (4)	0.512 (4)	0.482 (4)
MemoryOS (Kang et al., 2025a)	0.413	3	0.553 (3)	0.589 (3)	0.552 (3)
Nemori (Nan et al., 2025)	**0.502**	**1**	0.602 (2)	**0.781 (1)**	0.649 (2)
MAGMA (Jiang et al., 2026a)	0.467	2	**0.670 (1)**	0.741 (2)	**0.665 (1)**
SimpleMEM (Liu et al., 2026)	0.268	4	0.294 (5)	0.298 (5)	0.289 (5)
MemSkill (Zhang et al., 2026a)	0.082	6	0.221 (6)	0.220 (6)	0.127 (6)

While absolute scores fluctuate due to variations in grading strictness and prompt formulation, the comparative conclusions remain stable.

4.3.3 Discussion

Lexical metrics provide a convenient baseline but systematically diverge from semantic judgments due to two core failure modes: the Paraphrase Penalty, where correct abstractive answers are penalized for low token overlap, and the Negation Trap, where high overlap masks factual errors. Detailed examples are provided in Appendix F.

In contrast, the semantic judge demonstrates greater stability: architecture rankings remain consistent across different grading rubrics, suggesting it better reflects underlying memory quality rather than surface phrasing. Although absolute scores vary with prompt strictness and some models show rubric-aligned specialization, the relative ordering is robust.

Overall, these results support LLM-as-a-judge as a more reliable evaluation protocol for agentic memory, while highlighting the importance of careful prompt design.

4.4 Backbone Sensitivity and Format Stability

Agentic memory requires the backbone model to both answer queries and execute structured memory operations (e.g., updates and consolidation). Long-term stability thus depends on reliable adherence to strict output formats. To evaluate this "Stability Gap," we compare representative memory architectures using an API model (`gpt-4o-mini`) and an open-weight model (`Qwen-2.5-3B`).

Table 4 reveals a clear divergence driven by invalid structured outputs (e.g., malformed JSON, hallucinated keys) during memory maintenance: **1) Instruction Following vs. Reasoning:** While `Qwen-2.5-3B` demonstrates basic capability in conversational reasoning, it experiences a noticeable drop in End-Task Answer Scores and exhibits a significantly higher format error rate during memory updates compared to `gpt-4o-mini`. This "Silent Failure" implies that while the agent can converse fluently in the short term, its long-term memory becomes corrupted due to failed write operations. **2) Method Sensitivity:** The impact of the backbone varies by architecture complexity. Append-only systems are relatively robust, as they require minimal structured generation. In contrast, graph-based and episodic architectures are highly sensitive: extracting entities, constructing relations, and performing logical deduplication significantly increase format errors under weaker backbones, often leading to structural instability or collapse in memory maintenance.

Table 4: Backbone Sensitivity Analysis. Frequency of recoverable format deviations during memory operations is used. Higher values indicate greater reliance on fallback parsing due to inconsistent structured outputs.

Backbone	Method	Answer Score	Format Error
gpt-4o-mini	SimpleMem	0.289	1.20%
	Nemori	0.781	17.91%
Qwen-2.5-3B	SimpleMem	0.102	4.82%
	Nemori	0.447	30.38%

4.5 System Performance Evaluation

While accuracy is critical, the practical viability of agentic memory is constrained by latency and cost. Unlike read-only RAG systems, agentic memory introduces maintenance operations such as memory extraction, updates, and consolidation. We measure the user-facing load as retrieval (T_{read}), covering search and traversal, and generation (T_{gen}), including context processing and token decoding, while discussing maintenance overhead qualitatively.

In this section, we quantify user-facing latency ($T_{read} + T_{gen}$) and offline scalability using Table 5, while highlighting maintenance as a hidden system-level bottleneck.

Table 5: **The "Agency Tax": Efficiency Profiling.** We evaluate the trade-off between runtime user latency and offline construction cost. User Latency($T_{read} + T_{gen}$) dictates the interactive experience, while Construction Cost reflects the scalability and economic feasibility of the system. Note that Maintenance Cost is omitted as it is often handled asynchronously.

Method	User-Facing Latency (per turn)			Construction Cost (Offline)	
	Retrieval (T_{read})	Generation (T_{gen})	Total (s)	Time (h)	Tokens (k)
Full Context	N/A	1.726	1.726	N/A	N/A
LOCOMO (Maharana et al., 2024)	0.415	0.368	0.783	0.86	1,623
AMem (Xu et al., 2025c)	0.062	1.119	1.181	15.00	1,486
MemoryOS (Kang et al., 2025a)	31.247	1.125	32.372	7.83	4,043
Nemori (Nan et al., 2025)	0.254	0.875	1.129	3.25	7,044
MAGMA (Jiang et al., 2026a)	0.497	0.965	1.462	7.28	2,725
SimpleMem (Liu et al., 2026)	0.009	1.048	1.057	3.45	1,308
MemSkill (Zhang et al., 2026a)	0.005	0.301	0.306	0.60	1,796

4.5.1 Latency and Maintenance Trade-offs in MAG

We analyze the end-to-end user-perceived latency (T_{read}+T_{gen}) alongside the often-overlooked maintenance overhead (T_{write}). Although Full Context eliminates retrieval cost, it incurs the highest generation latency ($T_{gen} \approx 1.73$s), confirming that large pre-fill computation increases time-to-first-token. Lightweight systems such as SimpleMem and LOCOMO achieve sub-second latency (< 1.1s) through efficient indexing, while MAGMA maintains a balanced profile (~ 1.46s), adding modest overhead for graph traversal. In contrast, MemoryOS emerges as a clear bottleneck, with latency exceeding 32 seconds, suggesting that strict hierarchical paging (e.g., STM→LTM recursion) is impractical for interactive settings.

Beyond user-facing latency, maintenance introduces a hidden scalability constraint. Append-only or lightweight systems incur lower update cost, whereas structured architectures (e.g., MAGMA, AMem) require graph restructuring and LLM-driven consolidation. Although often asynchronous, excessive maintenance overhead risks throughput collapse, where updates lag behind user interactions and memory becomes stale. Thus, structured memory demands robust asynchronous infrastructure to remain viable at scale.

4.5.2 Offline Scalability: Time and Token Economics

Beyond online latency, we evaluate the offline cost of building the memory index. AMem requires approximately 15 hours for construction far slower than other baselines, suggesting super-linear update complexity (e.g., pairwise consolidation) that limits scalability on large datasets.

Token consumption further exposes cost trade-offs. Nemori uses over 7.04M tokens during index construction, nearly five times that of SimpleMem (1.3M). Although this yields strong accuracy, it reflects a substantial "intelligence tax," where improved memory quality incurs significantly higher operational cost. In comparison, MAGMA achieves a more favorable Pareto balance, delivering robust performance with moderate token usage (2.7M).

5 Conclusion and Future Directions

Our analysis shows that agentic memory is limited not only by architecture, but also by evaluation validity, scalability, and robustness.

1. Rethinking Benchmark and Evaluation Design. Future benchmarks should be saturation-aware. As context windows expand, full-context prompting may solve many tasks, making it harder to isolate the value of external memory. Benchmarks should therefore stress task volume, temporal depth, entity diversity, and long-range dependency, while using the Context Saturation Gap (Δ) as a diagnostic signal.

Evaluation should also move beyond lexical overlap. F1-style metrics often miss semantic correctness, while LLM-as-a-judge requires prompt calibration and robustness checks.

2. Designing Scalable and Robust Agentic Memory Systems. Agentic memory systems must balance accuracy, latency, cost, and reliability. Structured memory improves reasoning but introduces maintenance overhead, while lightweight approaches improve efficiency but may lack abstraction.

Future systems should explicitly model write latency, maintenance throughput, and user-facing

cost. They should also use backbone-aware memory operations, constrained decoding, validation layers, or adaptive schemas to reduce silent corruption.

Limitations

This survey has several limitations. First, agentic memory is evolving rapidly, so our taxonomy may miss concurrent or very recent systems. Second, our empirical analysis covers representative MAG architectures and selected benchmarks rather than all systems and settings; results may vary with implementations, prompts, model versions, and API behavior. These limitations call for broader, standardized evaluations of agentic memory systems.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Alen Adamyan, Tomáš Čížek, Matej Straka, Klara Janouskova, and Martin Schmid. 2025. Sam2rl: Towards reinforcement learning memory control in segment anything model 2. *arXiv preprint arXiv:2507.08548*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Linyue Cai, Yuyang Cheng, Xiaoding Shao, Huiming Wang, Yong Zhao, Wei Zhang, and Kang Li. 2025. A scenario-driven cognitive approach to next-generation ai memory. *arXiv preprint arXiv:2509.13235*.

Zouying Cao, Jiaji Deng, Li Yu, Weikang Zhou, Zhaoyang Liu, Bolin Ding, and Hai Zhao. 2025. Remember me, refine me: A dynamic procedural memory framework for experience-driven agent evolution. *arXiv preprint arXiv:2512.10696*.

Guoxin Chen, Zile Qiao, Xuanzhong Chen, Donglei Yu, Haotian Xu, Wayne Xin Zhao, Ruihua Song, Wenbiao Yin, Huifeng Yin, Liwen Zhang, and 1 others. 2025. Iterresearch: Rethinking long-horizon agents via markovian state reconstruction. *arXiv preprint arXiv:2511.07327*.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*.

Cody V Dong, Qihong Lu, Kenneth A Norman, and Sebastian Michelmann. 2025. Towards large language models with human-like episodic memory. *Trends in Cognitive Sciences*.

Xingbo Du, Loka Li, Duzhen Zhang, and Le Song. 2025a. MemR^3: Memory retrieval via reflective reasoning for LLM agents. *arXiv preprint arXiv:2512.20237*.

Yiming Du, Baojun Wang, Yifan Xiang, Zhaowei Wang, Wenyu Huang, Boyang Xue, Bin Liang, Xingshan Zeng, Fei Mi, Haoli Bai, and 1 others. 2025b. Memory-t1: Reinforcement learning for temporal reasoning in multi-session agents. *arXiv preprint arXiv:2512.20092*.

Wenzhe Fan, Ning Yan, and Masood Mortazavi. 2025. Evomem: Improving multi-agent planning with dual-evolving memory. *arXiv preprint arXiv:2511.01912*.

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, and 1 others. 2025a. Lightmem: Lightweight and efficient memory-augmented generation. *arXiv preprint arXiv:2510.18866*.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2025b. Memp: Exploring agent procedural memory. *arXiv preprint arXiv:2508.06433*.

Dongge Han, Camille Couturier, Daniel Madrigal Diaz, Xuchao Zhang, Victor Rühle, and Saravan Rajmohan. 2025. Legomem: Modular procedural memory for multi-agent llm systems for workflow automation. *arXiv preprint arXiv:2510.04851*.

Chuanrui Hu, Xingze Gao, Zuyi Zhou, Dannong Xu, Yi Bai, Xintong Li, Hui Zhang, Tong Li, Chong Zhang, Lidong Bing, and 1 others. 2026a. Evermemos: A self-organizing memory operating system for structured long-horizon reasoning. *arXiv preprint arXiv:2601.02163*.

Yuyang Hu, Jiongnan Liu, Jiejun Tan, Yutao Zhu, and Zhicheng Dou. 2026b. Memory matters more: Event-centric memory as a logic map for agent searching and reasoning. *arXiv preprint arXiv:2601.04726*.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, and 1 others. 2025. Memory in the age of ai agents. *arXiv preprint arXiv:2512.13564*.

Wei-Chieh Huang, Weizhi Zhang, Yueqing Liang, Yuanchen Bei, Yankai Chen, Tao Feng, Xinyu Pan, Zhen Tan, Yu Wang, Tianxin Wei, and 1 others.

2026. Rethinking memory mechanisms of foundation agents in the second half. *arXiv preprint arXiv:2602.06052*.

Yizhe Huang, Yang Liu, Ruiyu Zhao, Xiaolong Zhong, Xingming Yue, and Ling Jiang. 2025a. Memorb: A plug-and-play verbal-reinforcement memory layer for e-commerce customer service. *arXiv preprint arXiv:2509.18713*.

Zhengjun Huang, Zhoujin Tian, Qintian Guo, Fangyuan Zhang, Yingli Zhou, Di Jiang, Zeying Xie, and Xiaofang Zhou. 2025b. Licomemory: Lightweight and cognitive agentic memory for efficient long-term reasoning. *arXiv preprint arXiv:2511.01448*.

Yupeng Huo, Yaxi Lu, Zhong Zhang, Haotian Chen, and Yankai Lin. 2026. Atommem: Learnable dynamic agentic memory with atomic memory operation. *arXiv preprint arXiv:2601.08323*.

Rodrigo Toro Icarte, Richard Valenzano, Toryn Q Klassen, Phillip Christoffersen, Amir-massoud Farahmand, and Sheila A McIlraith. 2020. The act of remembering: A study in partially observable reinforcement learning. *arXiv preprint arXiv:2010.01753*.

Shian Jia, Ziyang Huang, Xinbo Wang, Haofei Zhang, and Mingli Song. 2025. Pisa: A pragmatic psych-inspired unified memory system for enhanced ai agency. *arXiv preprint arXiv:2510.15966*.

Zixia Jia, Jiaqi Li, Yipeng Kang, Yuxuan Wang, Tong Wu, Quansen Wang, Xiaobo Wang, Shuyi Zhang, Junzhe Shen, Qing Li, and 1 others. 2026. The ai hippocampus: How far are we from human memory? *arXiv preprint arXiv:2601.09113*.

Dongming Jiang, Yi Li, Guanpeng Li, and Bingzhe Li. 2026a. Magma: A multi-graph based agentic memory architecture for ai agents. *arXiv preprint arXiv:2601.03236*.

Dongming Jiang, Yi Li, Guanpeng Li, Qiannan Li, and Bingzhe Li. 2026b. Hage: Harnessing agentic memory via rl-driven weighted graph evolution. *arXiv preprint arXiv:2605.09942*.

Hanqi Jiang, Junhao Chen, Yi Pan, Ling Chen, Weihang You, Yifan Zhou, Ruidong Zhang, Yohannes Abate, and Tianming Liu. 2026c. Synapse: Empowering llm agents with episodic-semantic memory via spreading activation. *arXiv preprint arXiv:2601.02744*.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. 2025a. Memory os of ai agent. *arXiv preprint arXiv:2506.06326*.

Jikun Kang, Wenqi Wu, Filippos Christianos, Alex James Chan, Fraser David Greenlee, George Thomas, Marvin Purtorab, and Andrew Toulis. 2025b. Lm2: Large memory models for long context reasoning. In *Workshop on Reasoning and Planning for Large Language Models*.

Minki Kang, Wei-Ning Chen, Dongge Han, Huseyin A Inan, Lukas Wutschitz, Yanzhi Chen, Robert Sim, and Saravan Rajmohan. 2025c. Acon: Optimizing context compression for long-horizon llm agents. *arXiv preprint arXiv:2510.00615*.

Sangyeop Kim, Yohan Lee, Sanghwa Kim, Hyunjong Kim, and Sungzoon Cho. 2025. Pre-storage reasoning for episodic memory: Shifting inference burden to memory for personalized dialogue. *arXiv preprint arXiv:2509.10852*.

Taeyoon Kwon, Dongwook Choi, Hyojun Kim, Sunghwan Kim, Seungjun Moon, Beong-woo Kwak, Kuan-Hao Huang, and Jinyoung Yeo. 2025. Embodied agents meet personalization: Investigating challenges and solutions through the lens of memory utilization. *arXiv preprint arXiv:2505.16348*.

Chris Latimer, Nicoló Boschi, Andrew Neeser, Chris Bartholomew, Gaurav Srivastava, Xuan Wang, and Naren Ramakrishnan. 2025. Hindsight is 20/20: Building agent memory that retains, recalls, and reflects. *arXiv preprint arXiv:2512.12818*.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. 2024. A human-inspired reading agent with gist memory of very long contexts. *arXiv preprint arXiv:2402.09727*.

Haichang Li. 2025. Memory as a service (maas): Rethinking contextual memory as service-oriented modules for collaborative agents. *arXiv preprint arXiv:2506.22815*.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. 2025a. Hello again! llm-powered personalized agent for long-term dialogue. 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 5259–5276.

Kai Li, Xuanqing Yu, Ziyi Ni, Yi Zeng, Yao Xu, Zheqing Zhang, Xin Li, Jitao Sang, Xiaogang Duan, Xuelei Wang, and 1 others. 2026a. Timem: Temporal-hierarchical memory consolidation for long-horizon conversational agents. *arXiv preprint arXiv:2601.02845*.

Yi Li, Lianjie Cao, Faraz Ahmed, Puneet Sharma, and Bingzhe Li. 2026b. Hippocampus: An efficient and scalable memory module for agentic ai. *Preprint*, arXiv:2602.13594.

Zhiyu Li, Chenyang Xi, Chunyu Li, Ding Chen, Boyu Chen, Shichao Song, Simin Niu, Hanyu Wang, Jiawei Yang, Chen Tang, and 1 others. 2025b. Memos: A memory os for ai system. *arXiv preprint arXiv:2507.03724*.

Zouying Cao Li Yu, Jiaji Deng. 2025. Agentscopereme: Memory management kit for agents.

Shuai Ling, Lizi Liao, Dongmei Jiang, and Weili Guan. 2025. Memad: Structured memory of debates for enhanced multi-agent reasoning. In *Second Conference on Language Modeling*.

Genglin Liu, Shijie Geng, Sha Li, Hejie Cui, Sarah Zhang, Xin Liu, and Tianyi Liu. 2025a. Webcoach: Self-evolving web agents with cross-session memory guidance. *arXiv preprint arXiv:2511.12997*.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. 2026. Simplemem: Efficient lifelong memory for llm agents. *arXiv preprint arXiv:2601.02553*.

Jun Liu, Zhenglun Kong, Changdi Yang, Fan Yang, Tianqi Li, Peiyan Dong, Joannah Nanjekye, Hao Tang, Geng Yuan, Wei Niu, and 1 others. 2025b. Rcr-router: Efficient role-aware context routing for multi-agent llm systems with structured memory. *arXiv preprint arXiv:2508.04903*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Xinxin Liu, Weizhen Li, Weichen Sun, Xinlong Yang, Tianrui Qin, Xitong Gao, and Wangchunshu Zhou. 2025c. Compressed step information memory for end-to-end agent foundation models.

Yaoyao Liu, Bernt Schiele, and Qianru Sun. 2021. Rmm: Reinforced memory management for class-incremental learning. *Advances in neural information processing systems*, 34:3478–3490.

Jinghao Luo, Yuchen Tian, Chuxue Cao, Ziyang Luo, Hongzhan Lin, Kaixin Li, Chuyi Kong, Ruichao Yang, and Jing Ma. 2026. From storage to experience: A survey on the evolution of llm agent memory mechanisms.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*.

Wenyu Mao, Haosong Tan, Shuchang Liu, Haoyang Liu, Yifan Xu, Huaxiang Ji, and Xiang Wang. 2026. Bi-mem: Bidirectional construction of hierarchical memory for personalized llms via inductive-reflective agents. *arXiv preprint arXiv:2601.06490*.

Ali Modarressi, Ayyoob Imani, Mohsen Fayyaz, and Hinrich Schütze. 2023. Ret-llm: Towards a general read-write memory for large language models. *arXiv preprint arXiv:2305.14322*.

Hyungho Na, Yunkyeong Seo, and Il-chul Moon. 2024. Efficient episodic memory utilization of cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2403.01112*.

Jiayan Nan, Wenquan Ma, Wenlong Wu, and Yize Chen. 2025. Nemori: Self-organizing agent memory inspired by cognitive science. *arXiv preprint arXiv:2508.03341*.

Leyi Ouyang. 2025. Can memory-augmented llm agents aid journalism in interpreting and framing news for diverse audiences? *arXiv preprint arXiv:2507.21055*.

Siru Ouyang, Jun Yan, I Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T Le, Samira Daruki, Xiangru Tang, and 1 others. 2025. Reasoningbank: Scaling agent self-evolving with reasoning memory. *arXiv preprint arXiv:2509.25140*.

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

Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. 2025. Zep: a temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*.

Nazmus Sakib, Protoy Barai, Sifat Ishmam Parisa, and Anindya Iqbal. 2025. Memagent: A cache-inspired framework for augmenting conversational web agents with task-specific information.

Miao Su, Yucan Guo, Zhongni Hou, Long Bai, Zixuan Li, Yufei Zhang, Guojun Yin, Wei Lin, Xiaolong Jin, Jiafeng Guo, and 1 others. 2026. Beyond dialogue time: Temporal semantic memory for personalized llm agents. *arXiv preprint arXiv:2601.07468*.

Haoran Sun, Zekun Zhang, and Shaoning Zeng. 2025a. Preference-aware memory update for long-term llm agents. *arXiv preprint arXiv:2510.09720*.

Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. 2025b. Scaling long-horizon llm agent via context-folding. *arXiv preprint arXiv:2510.11967*.

Haoran Tan, Zeyu Zhang, Chen Ma, Xu Chen, Quanyu Dai, and Zhenhua Dong. 2025a. Membench: Towards more comprehensive evaluation on the memory of llm-based agents. *arXiv preprint arXiv:2506.21605*.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, and 1 others. 2025b. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8416–8439.

Xiangru Tang, Tianrui Qin, Tianhao Peng, Ziyang Zhou, Daniel Shao, Tingting Du, Xinming Wei, He Zhu, Ge Zhang, Jiaheng Liu, and 1 others. Agent kb: A hierarchical memory framework for cross-domain agentic problem solving. In *ICML 2025 Workshop on Collaborative and Federated Agentic Workflows*.

Dehao Tao, Guoliang Ma, Yongfeng Huang, and Minghu Jiang. 2026. Membox: Weaving topic continuity into long-range memory for llm agents. *arXiv preprint arXiv:2601.03785*.

He Wang, Wenyilin Xiao, Songqiao Han, and Hailiang Huang. 2025a. Stockmem: An event-reflection memory framework for stock forecasting. *arXiv preprint arXiv:2512.02720*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Yu Wang and Xi Chen. 2025. Mirix: Multi-agent memory system for llm-based agents. *arXiv preprint arXiv:2507.07957*.

Yu Wang, Ryuichi Takanobu, Zhiqi Liang, Yuzhen Mao, Yuanzhe Hu, Julian McAuley, and Xiaojian Wu. 2025b. Mem-{\alpha}: Learning memory construction via reinforcement learning. *arXiv preprint arXiv:2509.25911*.

Zixuan Wang, Bo Yu, Junzhe Zhao, Wenhao Sun, Sai Hou, Shuai Liang, Xing Hu, Yinhe Han, and Yiming Gan. 2025c. Karma: Augmenting embodied ai agents with long-and-short term memory systems. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE.

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.

Chao Wu and Gang Li. 2025. Enhancing generative agents in social simulations: Bridging memory, emotion, and governance for realistic social simulations. In *Intelligent Systems Conference*, pages 50–59. Springer.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2024. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*.

Yaxiong Wu, Yongyue Zhang, Sheng Liang, and Yong Liu. 2025a. Sgmem: Sentence graph memory for long-term conversational agents. *arXiv preprint arXiv:2509.21212*.

Zijun Wu, Yongchang Hao, and Lili Mou. 2025b. Tokmem: Tokenized procedural memory for large language models. *arXiv preprint arXiv:2510.00444*.

Derong Xu, Yi Wen, Pengyue Jia, Yingyi Zhang, Yichao Wang, Huifeng Guo, Ruiming Tang, Xiangyu Zhao, Enhong Chen, Tong Xu, and 1 others. 2025a. From single to multi-granularity: Toward long-term memory association and selection of conversational agents. *arXiv preprint arXiv:2505.19549*.

Haoran Xu, Jiacong Hu, Ke Zhang, Lei Yu, Yuxin Tang, Xinyuan Song, Yiqun Duan, Lynn Ai, and Bill Shi. 2025b. Sedm: Scalable self-evolving distributed memory for agents. *arXiv preprint arXiv:2509.09498*.

Jiexi Xu. 2025. Memory management and contextual consistency for long-running low-code agents. *arXiv preprint arXiv:2509.25250*.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025c. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Kristian Kersting, Jeff Z Pan, Hinrich Schütze, and 1 others. 2025. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Chang Yang, Chuang Zhou, Yilin Xiao, Su Dong, Luyao Zhuang, Yujing Zhang, Zhu Wang, Zijin Hong, Zheng Yuan, Zhishang Xiang, and 1 others. 2026a. Graph-based agent memory: Taxonomy, techniques, and applications. *arXiv preprint arXiv:2602.05665*.

Chengyuan Yang, Zequn Sun, Wei Wei, and Wei Hu. 2026b. Beyond static summarization: Proactive memory extraction for llm agents. *arXiv preprint arXiv:2601.04463*.

Hongkang Yang, Zehao Lin, Wenjin Wang, Hao Wu, Zhiyu Li, Bo Tang, Wenqiang Wei, Jinbo Wang, Zeyun Tang, Shichao Song, and 1 others. 2024b. $memory^3$: Language modeling with explicit memory. *arXiv preprint arXiv:2407.01178*.

Xiaofang Yang, Lijun Li, Heng Zhou, Tong Zhu, Xiaoye Qu, Yuchen Fan, Qianshan Wei, Rui Ye, Li Kang, Yiran Qin, and 1 others. 2026c. Toward efficient agents: Memory, tool learning, and planning. *arXiv preprint arXiv:2601.14192*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Yiqun Yao, Naitong Yu, Xiang Li, Xin Jiang, Xuezhi Fang, Wenjia Ma, Xuying Meng, Jing Li, Aixin Sun, and Yequan Wang. 2025. Egomem: Lifelong memory agent for full-duplex omnimodal models. *arXiv preprint arXiv:2509.11914*.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, and 1 others. 2025. Agentfold: Long-horizon web agents with proactive context management. *arXiv preprint arXiv:2510.24699*.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and 1 others. 2025. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*.

Qianhao Yuan, Jie Lou, Zichao Li, Jiawei Chen, Yaojie Lu, Hongyu Lin, Le Sun, Debing Zhang, and Xianpei Han. 2025. Memsearcher: Training llms to reason, search and manage memory via end-to-end reinforcement learning. *arXiv preprint arXiv:2511.02805*.

Dell Zhang, Yue Feng, Haiming Liu, Changzhi Sun, Jixiang Luo, Xiangyu Chen, and Xuelong Li. 2025a. Conversational agents: From rag to ltm. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pages 447–452.

Guibin Zhang, Muxin Fu, Guancheng Wan, Miao Yu, Kun Wang, and Shuicheng Yan. 2025b. G-memory: Tracing hierarchical memory for multi-agent systems. *arXiv preprint arXiv:2506.07398*.

Guibin Zhang, Muxin Fu, and Shuicheng Yan. 2025c. Memgen: Weaving generative latent memory for self-evolving agents. *arXiv preprint arXiv:2509.24704*.

Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. 2026a. Memskill: Learning and evolving memory skills for self-evolving agents. *arXiv preprint arXiv:2602.02474*.

Kai Zhang, Xinyuan Zhang, Ejaz Ahmed, Hongda Jiang, Caleb Kumar, Kai Sun, Zhaojiang Lin, Sanat Sharma, Shereen Oraby, Aaron Colak, and 1 others. 2025d. Assomem: Scalable memory qa with multi-signal associative retrieval. *arXiv preprint arXiv:2510.10397*.

Ningning Zhang, Xingxing Yang, Zhizhong Tan, Weiping Deng, and Wenyong Wang. 2026b. Himem: Hierarchical long-term memory for llm long-horizon agents. *arXiv preprint arXiv:2601.06377*.

Shengtao Zhang, Jiaqian Wang, Ruiwen Zhou, Junwei Liao, Yuchen Feng, Weinan Zhang, Ying Wen, Zhiyu Li, Feiyu Xiong, Yutao Qi, and 1 others. 2026c. Memrl: Self-evolving agents via runtime reinforcement learning on episodic memory. *arXiv preprint arXiv:2601.03192*.

Xin Zhang, Kailai Yang, Hao Li, Chenyue Li, Qiyu Wei, and Sophia Ananiadou. 2026d. Implicit graph, explicit retrieval: Towards efficient and interpretable long-horizon memory for large language models. *arXiv preprint arXiv:2601.03417*.

Yiran Zhang, Jincheng Hu, Mark Dras, and Usman Naseem. 2025e. Cogmem: A cognitive memory architecture for sustained multi-turn reasoning in large language models. *arXiv preprint arXiv:2512.14118*.

Yuxiang Zhang, Jiangming Shu, Ye Ma, Xueyuan Lin, Shangxi Wu, and Jitao Sang. 2025f. Memory as action: Autonomous context curation for long-horizon agentic tasks. *arXiv preprint arXiv:2510.12635*.

Xinkui Zhao, Qingyu Ma, Yifan Zhang, Hengxuan Lou, Guanjie Cheng, Shuiguang Deng, and Jianwei Yin. 2025. Ame: An efficient heterogeneous agentic memory engine for smartphones. *arXiv preprint arXiv:2511.19192*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19724–19731.

Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, and 1 others. 2025a. Memento: Fine-tuning llm agents without fine-tuning llms. *arXiv preprint arXiv:2508.16153*.

Sizhe Zhou and Jiawei Han. 2025. A simple yet strong baseline for long-term conversational memory of llm agents. *arXiv preprint arXiv:2511.17208*.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Jinhua Zhao, Bryan Kian Hsiang Low, and Paul Pu Liang. 2025b. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents. *arXiv preprint arXiv:2506.15841*.

Yuanjie Zhu, Liangwei Yang, Ke Xu, Weizhi Zhang, Zihe Song, Jindong Wang, and Philip S Yu. 2025. Llm-memcluster: Empowering large language models with dynamic memory for text clustering. *arXiv preprint arXiv:2511.15424*.

A Taxonomy of Agentic Memory

Figure 1 provides a comprehensive visual taxonomy of recent advancements in Memory-Augmented Generation (MAG). Given the rapid proliferation of memory architectures for LLM agents, this tree diagram categorizes contemporary literature into four distinct structural paradigms:

B Agentic Memory Background

B.1 Memory Operations in Agentic Systems

We characterize *agentic memory* as an external, non-parametric subsystem that interacts with an agent through two coupled processes: **inference-time recall** (reading memory to condition decisions) and **memory update** (writing, consolidating, and forgetting to maintain a useful long-term store). Let f_θ denote a frozen foundation model (or policy model) with parameters θ, and let $\mathcal{M}_t$ denote the external memory state at step t. Given an observation o_t (e.g., user input, tool output, sensor data) and agent state s_t (e.g., goals, plans, tool traces), the agent first produces a query q_t and recalls relevant memory:

$$q_t = \mathrm{Query}(o_t, s_t), \tag{3}$$
$$r_t = \mathrm{Read}(\mathcal{M}_t, q_t). \tag{4}$$

The retrieved content r_t is then integrated into the model input to produce an action or response:

$$a_t \sim \pi_\theta(o_t, r_t, s_t) \;=\; f_\theta\Big(\phi(o_t, s_t) \;\oplus\; \psi(r_t)\Big), \tag{5}$$

where $\phi(\cdot)$ formats current context, $\psi(\cdot)$ formats retrieved memory, and $\oplus$ denotes an integration operator (e.g., concatenation, schema-based slots, or cross-modal fusion). This abstraction makes explicit that agentic memory influences behavior through an external recall term rather than by updating θ.

Inference-time retrieval as approximate utility optimization. External memory retrieval typically selects items $\{m_i\}_{i=1}^N$ from $\mathcal{M}_t$ using a scoring function (dense similarity, sparse matching, or reranking). A common instantiation is top-k recall:

$$r_t = \mathrm{TopK}\Big(\{m_i\}_{i=1}^N;\ \mathrm{score}(q_t, m_i),\ k\Big). \tag{6}$$

However, in agentic settings, the ideal "relevance" is not purely semantic but *decision-conditional*. One can express an idealized retrieval objective as selecting memory that maximizes downstream utility:

$$r_t^\star = \arg\max_{r \subseteq \mathcal{M}_t} \mathbb{E}\big[\, U(a_t \mid o_t, r, s_t) \,\big], \tag{7}$$

where $U(\cdot)$ denotes agent utility (e.g., task success, efficiency, robustness). Practical systems approximate (7) using similarity search, learned rerankers, multi-hop retrieval, planner-guided recall, or retrieval policies trained to better align $\mathrm{score}(\cdot)$ with utility.

Memory update as explicit memory actions. After producing an action a_t (and possibly observing its outcome), the agent updates external memory through a write function:

$$\mathcal{M}_{t+1} = \mathrm{Write}(\mathcal{M}_t, o_t, a_t, s_t). \tag{8}$$

It is often useful to make updates explicit as *memory actions*. Let $u_t \in \mathcal{U}$ denote a memory action such as STORE, UPDATE, SUMMARIZE, LINK, EVICT, or DELETE. Then:

$$u_t = g(o_t, a_t, s_t), \qquad \mathcal{M}_{t+1} = T(\mathcal{M}_t, u_t), \tag{9}$$

where $g(\cdot)$ may be rule-based, model-driven, or learned, and T applies the chosen action to the memory store. This view connects naturally to RL-guided memory, where $g(\cdot)$ can be optimized as a policy over memory actions.

C Related Work

Several recent surveys have examined memory mechanisms for Agentic AI systems, each from a distinct vantage point. The AI Hippocampus (Jia et al., 2026) presents a broad synthesis organized around a brain-inspired trichotomy of implicit memory, explicit memory, and agentic memory, further extending the analysis to multimodal settings involving vision, audio, and embodied interaction. Memory in the Age of AI Agents (Hu et al., 2025) proposes a "forms–functions–dynamics" framework that categorizes agent memory along three orthogonal axes: architectural form, functional role, and lifecycle dynamics, providing a comprehensive conceptual vocabulary for the rapidly fragmenting landscape of agent memory research. Toward Efficient Agents (Yang et al., 2026c) shifts the focus from architectural expressiveness to deployment cost, surveying efficiency-oriented techniques across three core agent components: memory, tool learning, and planning. In addition, this

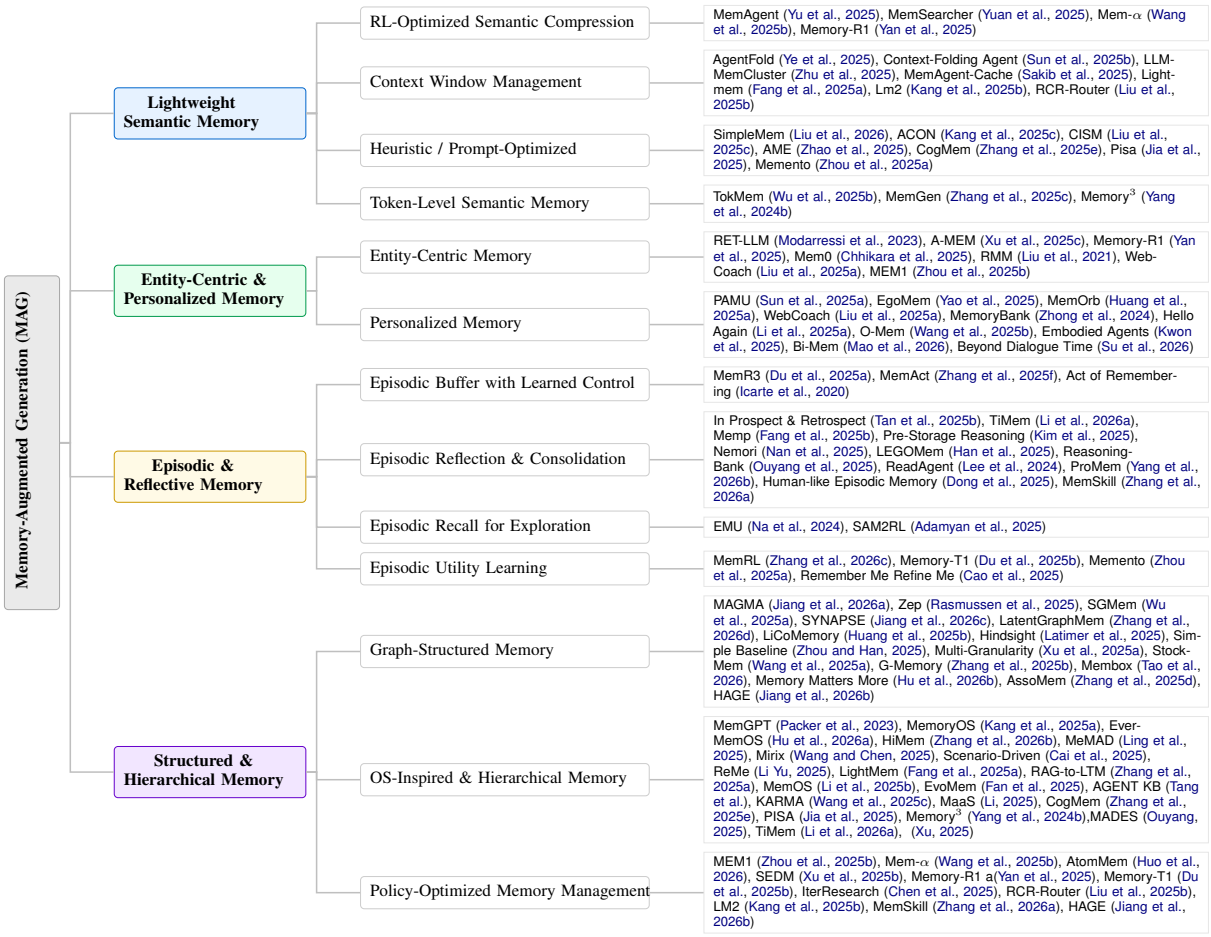

Figure 1: Taxonomy of Memory-Augmented Generation (MAG) systems.

survey discusses compression, context management, and reinforcement-learning-based reward design as shared principles to reduce latency, token consumption, and interaction steps. More recently, Rethinking Memory Mechanisms (Huang et al., 2026) assembles a large-scale survey of over 200 papers, organizing memory along three dimensions: substrate, cognitive mechanism, and subject, while reviewing learning policies over memory operations and cataloguing existing evaluation benchmarks. From Storage to Experience (Luo et al., 2026) offers an evolutionary perspective, formalizing memory development into three progressive stages: storage, reflection, and experience. It also identifies long-range consistency, dynamic environments, and continual learning as the core drivers of this evolution. Graph-based Agent Memory (Yang et al., 2026a) narrows the scope to graph-based memory paradigms of knowledge graphs, temporal graphs, hypergraphs, and hierarchical trees. In addition, it systematically analyzes extraction, storage, retrieval, and evolution along the memory lifecycle.

D Prompt Library

This section details the prompt templates used for all experimental evaluations. To ensure reproducibility, we provide the specific instructions for memory construction, query processing, and the varying sensitivities of our evaluation protocols.

To provide a structured comparison, we classify the prompt designs into three operational stages: Memory Construction (Build), Query Processing (Query), and Response Generation (Answer). Table 6 summarizes the design patterns across the evaluated systems.

D.1 Memory Construction and Retrieval

Different memory architectures require different construction strategies. This section outlines the prompts used by MAG systems to consolidate raw interaction history into long-term storage and refine user queries.

Build Prompts (Memory Indexing) Used by MAG systems to consolidate raw interaction history into long-term storage. The evaluated systems

Table 6: Taxonomy of System Operation Prompts across Memory Architectures.

System	Build Strategy (Memory Construction)	Query Strategy	Answer Strategy (Synthesis)
MemoryOS	Profile-based (User profiling, Knowledge extraction)	N/A (Direct Semantic Search)	Role-playing & Profile-enriched
AMem	Flat/Turn-based (Content analysis)	LLM Keyword Extraction	Retrieved Memory Context
Nemori	Episodic (Boundary detection, Episode generation)	N/A (Direct Semantic Search)	Episode-based Retrieval
MAGMA	Graph-based (Event extraction, Multi-hop reasoning)	Multi-hop Entity Parsing	Graph Traversal Synthesis
SimpleMem	Minimalist/Turn-level	Keyword Generation	Context-based

utilize distinct structural representations:

- **Profile Based (MemoryOS):** Instructs the LLM to extract observable user traits and merge them into an evolving profile.
- **Episodic (Nemori):** Segments continuous dialogue into discrete episodes using boundary detection.
- **Graph Based (MAGMA):** Translates interactions into relational structures (e.g., event extraction, triplet arrays).

[PLACEHOLDER: Insert Build Prompt for memory consolidation, e.g., "Summarize the following interaction into atomic facts..."]

Query Refinement Prompts While many systems (like Nemori and MemoryOS) bypass explicit refinement in favor of direct semantic search algorithms, systems like AMem and SimpleMem use LLMs to transform user queries into optimized search vectors or keywords.

[PLACEHOLDER: Insert Query Prompt, e.g., "Given the conversation history, rewrite the user query for better retrieval..."]

D.2 Response Generation

Answer Generator Prompts The standard templates used by all baselines (RAG, MAG, and Full-Context) to produce final responses based on retrieved or provided context. The prompt designs vary based on the context strategy (e.g., profile-enriched role-playing for MemoryOS, episodic retrieval for Nemori) and specific constraints (e.g., temporal awareness or multi-hop reasoning for AMem and MAGMA).

[PLACEHOLDER: Insert Answer Generator Prompt, e.g., "You are an assistant with access to the following memory shards. Answer the question based on..."]

D.3 LLM-as-a-Judge Evaluation Protocols

We utilize `gpt-4o-mini` as our backbone judge. To comprehensively evaluate architecture performance across the diverse grading criteria mentioned in Section 4.3.2, we structure our evaluation into two categories: literature-derived baseline prompts and sensitivity rubrics.

D.3.1 Literature-Derived Baselines

These prompts represent different community standards for "correctness" and are sourced directly from existing benchmarks.

Prompt 1: MAGMA (Semantic Correctness & Context) Derived from the MAGMA framework (Jiang et al., 2026a), this multi-level scoring protocol prioritizes information integration and reasoning. It emphasizes interpersonal knowledge retrieval and semantic equivalence, with specific guidelines for temporal and factual preservation.

```
Score the answer on a scale from 0.0 to 1.0 based on semantic correctness.
Scoring Scale:
- 1.0: Perfect match - contains all key information, semantically equivalent
- 0.8: Mostly correct - captures main point but may have minor differences
- 0.6: Partially correct - has some correct info but incomplete
- 0.4: Somewhat related - touches on topic but misses significant info
- 0.2: Barely related - answer is mostly incorrect
- 0.0: Completely wrong - answer is unrelated or contradicts gold answer
Instruction:
Focus on user-interpersonal knowledge and temporal generosity.
Focus on semantic equivalence, not exact wording.
Assign partial credit for partially correct answers.
Input:
Question: {question}
Gold answer: {gold_answer}
Generated answer: {generated_answer}
Output (JSON):
{ "score": 1.0, "reasoning": "..." }
```

Prompt 2: Nemori (Generous Semantic Matching) Adapted from the Nemori paper (Nan et al.,

2025), this is a lenient, semantics-oriented evaluation scheme. It emphasizes entity recall and judges whether the generated answer captures the same underlying concept as the ground truth using a binary (CORRECT/WRONG) classification, tolerating paraphrasing and verbosity.

```
Your task is to label an answer as CORRECT or WRONG.
You will be given:
(1) a question
(2) a gold (ground truth) answer
(3) a generated answer
Evaluation Guidelines:
- Be generous in grading.
- If the generated answer conveys the same meaning or topic as the gold answer, mark it as CORRECT.
- Ignore differences in wording, phrasing, or length.
- Accept paraphrases and semantically equivalent answers.
- For time-related questions, accept different formats (e.g., "May 7" vs "7 May").
Input:
Question: {question}
Gold Answer: {gold_answer}
Generated Answer: {generated_answer}
First, provide a one-sentence reasoning, then output the result.
Output (JSON):
{ "label": "CORRECT" } or { "label": "WRONG" }
```

Prompt 3: SimpleMem (Relevance & Accuracy) Adapted from the SimpleMem baseline (Liu et al., 2026), this prompt focuses on retrieval precision and core fact preservation. It explicitly balances relevance, factual grounding, and tolerance to representational variation via a *Robustness Protocol*.

```
You are an expert Relevance & Accuracy Evaluator.
Your task is to determine whether the Predicted Answer successfully retrieves the necessary information to answer the Question, based on the Reference Answer.
Input:
Question: {question}
Reference Answer: {reference}
Predicted Answer: {prediction}
Evaluation Criteria:
1. Responsiveness to Query
The predicted answer must directly address the specific question and remain topically aligned with the user's intent.
2. Core Fact Preservation
The prediction must capture the key signal or core entity from the reference (e.g., who, what, or outcome).
3. Informational Utility
The answer must provide meaningful value. Even if concise, it should convey the essential information required by the question.
4. Robustness Protocol (Acceptable Variances)
You must treat the following variations as valid matches:
- Temporal & numerical tolerance (e.g., ±1–2 days, rounded numbers)
- Granularity differences (e.g., "Afternoon" vs. "14:05", "Late October" vs. "Oct 25")
- Information subsetting (partial but sufficient answers)
- Synonymy and format variation
Grading Logic:
- Score 1.0 (Pass): Contains relevant core information OR satisfies robustness conditions above.
- Score 0.0 (Fail): Missing core information, irrelevant, or fails to answer the question.
Output Format (JSON only):
{
"score": 1.0,
"reasoning": "Brief explanation focusing on relevance and core match."
}
```

E Baseline Configurations

This section details the hyper-parameter settings and model versions for the evaluated architectures. To ensure fair and objective comparisons, we strictly follow the default configuration settings provided in their respective open-source repositories, with the following standardized modifications applied across all baseline systems:

- **Embedding Model:** All dense retrieval operations are uniformly configured to use `all-MiniLM-L6-v2` (Wang et al., 2020), replacing any system-specific default embedding models to ensure a controlled baseline for semantic matching.

- **LLM Temperature:** The generation temperature is fixed at 0.3 across all backbone LLMs to maintain a consistent balance between determinism and reasoning capability.
- **Retrieval Top-k:** For final answer synthesis that relies on retrieving raw conversation history or memory chunks, we uniformly set the retrieval scope to top-$k = 10$.
- **Max Tokens:** The maximum token limits for generation and context windows are maintained at their repository-specific defaults to respect the intended design of each architecture.

A summary of these unified hyper-parameters alongside the specific operational parameters for each evaluated system is provided in Table 7.

F Case Studies: Why Lexical Metrics Fail

To further investigate the ranking discrepancies observed in Section 4.3.2, we conduct a qualitative analysis of representative cases where lexical metrics (e.g., F1) disagree with semantic judgments. Rather than presenting isolated examples, we organize these cases into a set of recurring failure mechanisms that reflect inherent limitations of token level evaluation.

We identify four common patterns:

- **Surface Variation:** Correct answers expressed with additional context or alternative phrasing are penalized due to reduced lexical overlap.
- **Semantic Equivalence Gap:** Equivalent meanings conveyed through different formats or synonyms result in zero or near-zero scores.
- **Polarity Flip:** Minor lexical changes (e.g., negation) invert the semantic meaning while preserving high token overlap.
- **Entity Drift:** Incorrect entities or values are substituted within otherwise similar sentence structures, leading to inflated lexical similarity despite factual errors.

Table 9 presents representative examples illustrating these failure modes. These cases demonstrate that lexical metrics are not merely noisy, but systematically misaligned with the abstraction, normalization, and reasoning behaviors exhibited by modern LLM-based systems.

These observations complement the quantitative findings in Section 4.3.2 and suggest a fundamental mismatch: lexical metrics operate on surface form alignment, whereas agentic systems increasingly rely on abstraction, normalization, and compositional reasoning. As a result, improvements in reasoning quality may not be reflected and can even be penalized by traditional token-based evaluation.

Table 7: Key hyper-parameter configurations for the evaluated memory architectures. To ensure fair comparison, embedding models, LLM generation temperatures, and final answer retrieval scopes are strictly standardized across all baselines, while structural capacity parameters (e.g., max tokens) follow repository defaults.

Method	Embedding Model	LLM Temp.	Final Answer Top-k	System-Specific Defaults (Max Tokens & Structure)
Full Context	N/A	0.3	N/A	Max Tokens: 128k (`gpt-4o-mini`)
LOCOMO	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Buffer Size: Default
AMem	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Keyword Extractor Temp: Default
MemoryOS	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Update Frequency: Default
Nemori	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Boundary Temp: 0.1
MAGMA	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Consolidation Threshold: Default
SimpleMem	`MiniLM-L6-v2`	0.3	10	Max Tokens: Default; Synthesis Strategy: Default

Table 8: Overview of memory systems and experimental configurations. We use `gpt-4o-mini` as the primary controller for all methods in the main benchmark to normalize reasoning costs.

Method	Memory Structure	Update Policy	Retrieval Scope
A-MEM	Linked Node Graph (Atomic Units + Tags)	Evolutionary: LLM-based node rewriting & dynamic linking	Dense Embedding Similarity (Top-k)
MemoryOS	Hierarchical Tiers (STM $\rightarrow$ LTM)	Rule-based: Frequency/Recency-based promotion	Cascading Hierarchy Search
MAGMA	Multi-relational Graph + Vector Index	Asynchronous: Dual-stream consolidation (Long-term)	Intent-guided Subgraph Traversal
Nemori	Dual Memory (Episodic Tree + Semantic Graph)	Gradient-inspired: Contextual memory modification	Hybrid (Top-k Episodes + Semantic Facts)
SimpleMem	Hybrid Index (Dense/Sparse) of Compressed Units	Synchronous: On-the-fly synthesis & deduplication	Planner-guided Multi-view Search

Table 9: Mechanism Oriented Failure Cases of Lexical Metrics. Lexical scores (F1) are contrasted with semantic judgments to highlight systematic mismatches.

Query & Gold Truth	Model Answer	Failure Type	F1	Judge	Analysis
Q: What is the duration of the event? **Gold:** 18 days	The total duration was 18 days.	Surface Variation	0.50	1.00	Additional phrasing lowers lexical precision despite identical meaning.
Q: What time does the event start? **Gold:** 14:00	2 PM	Semantic Equivalence Gap	0.00	1.00	Equivalent time representations yield zero token overlap.
Q: Describe the price level. **Gold:** cheap	inexpensive	Semantic Equivalence Gap	0.00	1.00	Synonym substitution is not captured by lexical matching.
Q: Is the software compatible with Mac? **Gold:** compatible with Mac	**not** compatible with Mac	Polarity Flip	0.857	0.00	Negation reverses meaning while preserving token overlap.
Q: Who completed the project? **Gold:** John completed the project	Sarah completed the project	Entity Drift	0.75	0.00	Incorrect entity maintains structure but changes semantics.
Q: How many items were included? **Gold:** three items	five items	Entity Drift	0.50	0.00	Numerical substitution is partially rewarded due to shared tokens.