# Survey on AI Memory: Theories, Taxonomies, Evaluations, and Emerging Trends

TING BAI[1,2]*, JIAYANG FAN[1,2], XIAOSHUAI WEN[1,2], JIAZHENG KANG[1,2], HENGZHI LAN[1,2], RUOCHEN ZHAO[1,2], PINGZHENG WU[1,2], ZEPENG ZHANG[1,2], YUTIAN ZHONG[1,2], GEZI LI[3], DONGYIN LIN[3],

[1] BaiJia AI Team,

[2] Beijing University of Posts and Telecommunications, Beijing, China,

[3] Huawei Technologies Co., Ltd.

Memory is a cornerstone of cognitive capability in AI systems, empowering them with dynamic adaptation, complex reasoning, and experiential learning. With the advent of LLM-driven agents, memory architectures have evolved from simple context windows to sophisticated systems that integrate parametric and non-parametric storage, fuse multimodal information, and facilitate knowledge sharing within multi-agent ecosystems. Despite these advancements, existing literature remains fragmented, often lacking integrative perspectives rooted in cognitive psychology to bridge computational mechanisms with human-like memory processes. Furthermore, there is a scarcity of unified taxonomies and systematic evaluation frameworks, particularly regarding collective learning and coordinated reasoning in multi-agent systems. To address these gaps, this survey presents a comprehensive overview of AI memory mechanisms anchored in a unified theoretical framework. We propose a structured "4W Memory Taxonomy" to enable consistent analysis across diverse architectures. Building on this foundation, we systematically review memory systems in both single- and multi-agent contexts, examining their architectures, functions, applications, and evaluation methodologies. By synthesizing cognitive theories with engineering benchmarks, this work provides a coherent roadmap for advancing the theoretical understanding and technological development of AI memory. An open-access resource is available at https://github.com/BAI-LAB/Survey-on-AI-Memory.

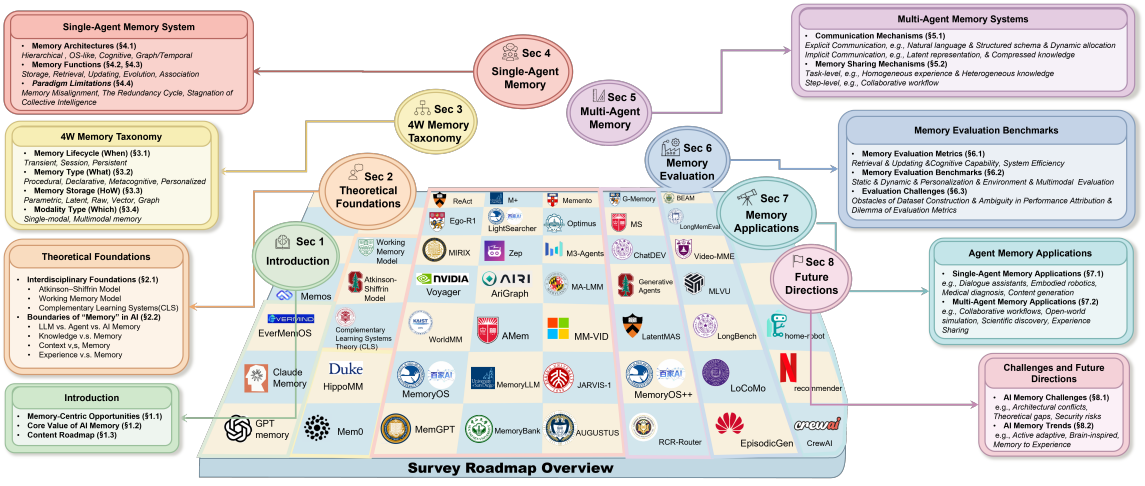


Fig. 1. The Evolutionary Landscape of AI Memory: A Content Roadmap of the Survey.

*Corresponding author: Ting Bai.

## 1 Introduction

Large Language Model (LLM)-driven agents have emerged as a pivotal bridge connecting advanced AI capabilities with complex real-world scenarios. These systems are facilitating a wide array of sophisticated applications, ranging from intelligent decision-making in healthcare and autonomous collaboration in software engineering to environmental perception in embodied robotics and the development of collective intelligence in open-world simulations [89, 148]. Currently, intelligent agents are undergoing a paradigm shift, transitioning from efficient single-task execution to continuous adaptation, evolving capabilities, and the accumulation of experiences within dynamic environments. In this context, AI memory acts as a cornerstone, empowering agents to maintain behavioral coherence, make rational decisions, and collaborate effectively. Unlike traditional AI systems constrained by task isolation and stateless interactions, which prevent historical data reuse. LLM-powered agents can overcome these limitations through sophisticated, structured memory architectures [168]. The strategic integration of memory mechanisms offers a transformative opportunity to bridge the gap between reactive processing and proactive intelligence.

### 1.1 Opportunities Toward Memory-Centric AI

Large Language Models (LLMs) have solidified their status as the core pillar of the AI field, owing to their exceptional capabilities in natural language understanding, generation, and reasoning, completely reshaping the paradigm for constructing AI systems [148, 193]. Building upon this foundation, LLM-driven agents possess greater flexibility and environmental adaptability. They can independently complete complex task planning, tool calling, and multi-step reasoning through dynamic interactions with the external environment [161]. For instance, in healthcare, they assist in medical diagnosis by analyzing patient histories; in finance, they optimize investment strategies based on market trends; in education, they provide personalized learning guidance. These applications drive AI toward more advanced autonomous intelligence and human-centered design [89]. While LLMs demonstrate remarkable proficiency in mastering linguistic rules, world knowledge, and task paradigms from massive textual data [193], standard LLMs encounter two key bottlenecks. First, despite advancements, the inherent constraints of context windows hinder the processing of ultra-long texts and the maintenance of extensive cross-session interactions. Second, the absence of mechanisms for the effective accumulation and reuse of historical memory renders standard LLMs inherently stateless. They treat each interaction as an isolated event, failing to leverage past information to optimize subsequent decisions [19]. Consequently, these limitations impede performance across diverse scenarios: conversational assistants fail to retain user preferences across sessions; embodied agents struggle to accumulate exploration experience for motion planning; and multi-agent systems exhibit inefficient collaboration due to the inability to share experiences [68, 188].

The memory mechanism provides a critical pathway to address these bottlenecks. As a core component of AI agents, memory enables AI systems to retain historical interaction information, store contextual data, and optimize future behaviors based on historical memory—thus achieving a leap from "one-time task execution" to "sustained autonomous evolution". For example, OpenAI's ChatGPT launched its Memory feature in 2024, which memorizes user preferences and cross-session interaction history, enhances the accuracy of personalized responses, and validates memory's core value in commercial products. The open-source community has also advanced rapidly [24, 73, 124]: the MemoryOS framework [73] pioneers the concept of a memory operating system, supports multiple LLM platforms, enables centralized management of agent memory, and offers a unified abstraction for memory storage, updating, retrieval, and response generation. These practices confirm that memory mechanisms are the core support for AI agents in achieving personalization, continuity, and autonomy, bridging AI technology with complex real-world demands [24].

### 1.2 Core Value of AI Memory

The core value of AI memory transcends the mere mitigation of technical bottlenecks in large language models (LLMs), such as limited context windows and stateless interactions [19, 97]. Instead, it functions as a transformative enabler, elevating AI systems from generalized tools to adaptive, collaborative, and human-centric intelligent agents. This shift drives fundamental leaps in capability and practical utility, manifesting through a layered progression from foundational identity to evolutionary intelligence.

Foundational to this transformation is the role of memory in agentification, which drives the transition of LLMs from passive "tools" to active "subjects." Memory serves as the prerequisite for models to evolve into persistently adaptive agents [161]. In its absence, AI systems lack temporal continuity and cognitive consistency, treating each interaction as an isolated task and failing to accumulate experience [89]. By equipping agents to encode, retrieve, and utilize historical information, memory enables the formation of user profiles [89, 197], the retention of task execution trajectories [131], and the maintenance of role consistency [116]. For instance, mechanisms like ChatGPT Memory retain cross-session preferences, converting the system from a static interface into a "user-aware" assistant. This evolution from stateless responsiveness to persistent existence underpins the very definition of autonomous agents [168].

Building upon this persistent identity, memory acts as an amplifier of advanced capabilities that directly bridges the gap between general model potential and scenario-specific pragmatic value. It facilitates experience reuse across tasks, significantly reducing redundant trial-and-error [81, 131] while enabling systems like MemoryOS [73] to standardize storage for personalized interaction and user alignment. Similarly, by leveraging memory to store self-critiques and intermediate logic, models achieve coherent long-horizon reasoning [131, 146], substantially improving performance on complex benchmarks [132]. Crucially, this capacity for maintaining workflow coherence translates directly to real-world utility, supporting multi-step execution in domains ranging from software development [119] to medical diagnosis [25] and scientific discovery [44].

Ultimately, AI memory functions as the engine of intelligent evolution, bridging human-like cognition and the pursuit of Artificial General Intelligence (AGI). It lays the foundation for lifelong learning and human-aligned interaction [168]. From an evolutionary standpoint, memory supports continuous self-improvement through experience abstraction, error correction, and knowledge updating. Cognitively, architectures inspired by human memory mechanisms render AI interactions more intuitive. By emulating core human memory features, these designs align AI systems with human cognitive patterns—a critical step, as lifelong learning and adaptive cognition are central to achieving general intelligence [32, 89]. AI memory is not merely a storage module but a dynamic cognitive substrate that integrates technical functionality, practical utility, and cognitive alignment. Its core value lies in endowing agents with persistent identity, amplified capabilities, scenario-specific utility, and evolutionary potential, making it an indispensable component in the era of LLM-driven intelligent systems.

### 1.3 Content Roadmap of the Survey

This survey contributes by establishing a comprehensive taxonomy and integrating theoretical frameworks for AI memory. The content roadmap of this survey is presented in **Fig. 1**. Subsequent sections are logically organized to support a holistic understanding of AI memory mechanisms: **Section 2** establishes the theoretical foundation by synthesizing cognitive psychology and neuroscience models into actionable design patterns for AI agent memory, while clarifying the conceptual boundaries of "memory" in AI; building on this, **Section 3** introduces the *4W Memory Taxonomy* (i.e., When-What-How-Which) to systematically categorize memory mechanisms; **Section 4** and **Section 5**

survey representative memory architectures and key technical components in single-agent and multi-agent systems, respectively, including core operations, advanced capabilities, and practical limitations; **Section 6** reviews evaluation dimensions for AI memory, covering performance metrics, representative benchmarks, and prevalent evaluation paradigms as well as common challenges in assessment; **Section 7** analyzes real-world applications, highlighting AI memory as a core enabler for improving agent performance across diverse domains; **Section 8** discusses open challenges and future research trends; finally, **Section 9** concludes the survey by synthesizing key findings and outlining future prospects for AI memory mechanisms.

## 2 Theoretical Foundations of AI Memory

In this section, we establish the conceptual bedrock for memory-augmented agents by bridging biological principles with computational definitions. To provide a rigorous theoretical framework for the subsequent analysis, we structure the discussion into two primary dimensions:

- **Interdisciplinary Foundations (§2.1)**: We introduce two cognitive psychology models, the Atkinson–Shiffrin Tri-Store Model and the Working Memory Model, and one cognitive neuroscience theory, the Complementary Learning Systems Theory (CLS), all of which have inspired the design philosophies of AI memory systems, deriving three essential architectural patterns: the separation of index from content, phasic consolidation, and structured workspace coordination.
- **Boundaries of "Memory" in AI (§2.2)**: We first clarify the scope of AI memory, encompassing the concrete manifestations of AI Memory vs. Agent Memory vs. LLM Memory. Furthermore, we rigorously distinguish AI memory from related but distinct concepts: static knowledge (Memory vs. Knowledge), ephemeral context windows (Memory vs. Context), and abstracted experience (Memory vs. Experience).

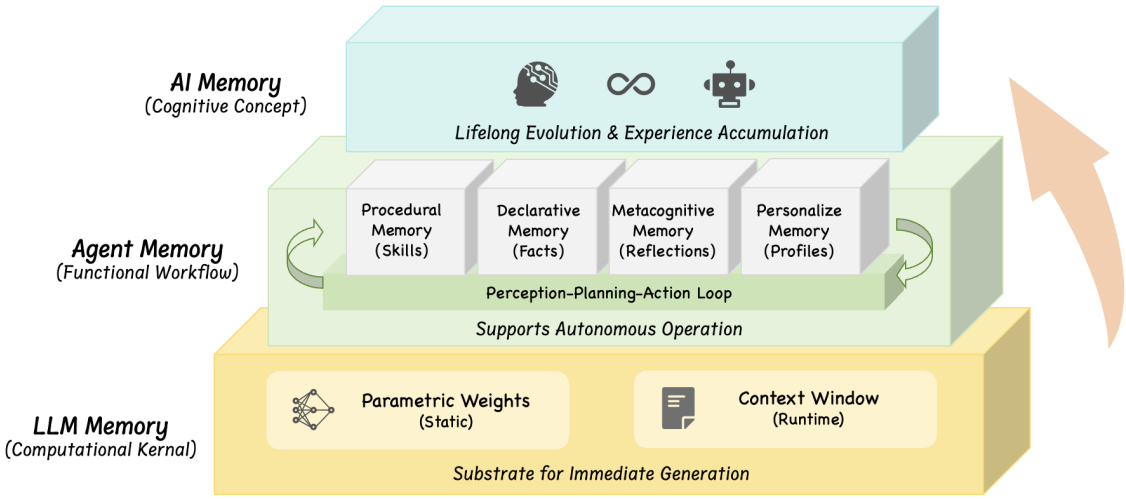


Fig. 2. An AI Boundary to Clarify the Interrelationships Among LLM Memory, Agent Memory, and AI Memory. LLM memory forms the low-level computational kernel for prediction. Agent memory provides the functional workflow to support autonomy and the execution of complex tasks via perception-planning-action loops. AI memory represents the overarching cognitive concept aimed at lifelong evolution, long-term persistence, and adaptation.

### 2.1 Interdisciplinary Foundations

This section grounds agent memory in foundations from cognitive psychology and neuroscience, including the Atkinson–Shiffrin Tri-Store Model, which delineates the macro-flow from sensory to long-term storage, and the Working Memory Model, which details the multicomponent nature of the active workspace. Cognitive neuroscience offers the Complementary Learning Systems Theory, explaining the synergy between the rapid hippocampal indexing and the slow cortical consolidation. Collectively, these frameworks motivate three design patterns: separating index from content, multiphase consolidation, and structured workspace coordination.

*2.1.1 Atkinson–Shiffrin Tri-Store Model.* The classic Atkinson–Shiffrin model [5] conceptualizes human memory as three interacting stores: sensory registers, a short-term store of limited capacity, and a durable long-term store. In brief: sensory memory is an ultra-short buffer for each sense, with iconic memory for vision that lasts roughly half a second and echoic memory for sound that persists for a few seconds [28, 135], capturing a high-fidelity snapshot that makes perception feel continuous; short-term or working memory is a focused mental workspace that can actively hold only a handful of items, about four meaningful chunks [26], for a few seconds unless we keep them alive by rehearsing or grouping them; long-term memory is the vast library of what we know and have lived, including facts, personal events, and skills, organized by meaning and associations and capable of lasting from days to decades.

These stores are coordinated by control processes including attention, rehearsal, and retrieval [5]. Sensory memory first captures raw input in modality-specific registers; selective attention then lifts a small subset into short-term memory, where it can be kept active for a few seconds and reorganized by rehearsal and chunking; through maintenance rehearsal and elaborative encoding that connect new material to existing knowledge, part of this content is stored in long-term memory; later, retrieval cues draw information from long-term memory back into short-term memory to guide thinking and action, with successful use strengthening associations and failures prompting renewed encoding. In summary, the model portrays memory as a sequence of stages: brief sensory traces, a limited-capacity short-term workspace, and an enduring long-term store, connected by controllable processes.

*2.1.2 Working Memory Model.* While the Atkinson–Shiffrin model treats short-term storage as a unitary system, Baddeley and Hitch argued that this view could not explain how we manipulate information during complex tasks [9]. They reframed the short-term store as a multicomponent working memory system. In this framework, a central executive acts as a capacity-limited controller that directs attention and coordinates resources rather than merely storing data. It is supported by two domain-specific subsystems: the phonological loop, which maintains verbal material through rehearsal, and the visuospatial sketchpad, which holds visual and spatial representations. This modular design implies that performance relies on how task demands map onto these components, reducing interference when simultaneous tasks draw on distinct subsystems rather than competing for a single short-term workspace.

Baddeley introduced the episodic buffer to address limitations of the original multicomponent model, particularly its inability to explain how separate subsystems interact with the vast long-term library [7, 8]. This fourth component serves as a temporary interface that integrates information from the phonological loop, the visuospatial sketchpad, and long-term memory into a unified, multimodal episode. Unlike the specialized subsystems, the buffer is capable of binding distinct features into coherent representations that are accessible to conscious awareness. This addition bridges the gap between fluid working memory processes and crystallized long-term knowledge, allowing the central executive to manipulate integrated episodes for learning, reasoning, and planning.

*2.1.3 Complementary Learning Systems Theory.* As a theory in Computational Neuroscience, the complementary learning systems theory [106] conceptualizes the brain's memory architecture as a synergistic partnership between the hippocampus and the neocortex. The hippocampus functions as a nimble encoder and indexer that rapidly binds and tags the pieces of a new experience for later reinstatement. The neocortex acts as deep storage that holds content and updates gradually to protect existing knowledge. This complementary division of hippocampus and neocortex lets new episodes be captured quickly without overwriting what is already known. When something novel happens, synapses adjust like finely tuned knobs, increasing the ease with which certain pathways fire again. In the hippocampus, this rapid plasticity helps encode the episode by linking its elements and by keeping similar experiences distinct, which gives fresh memories an immediate foothold and leaves a fragile but retrievable trace ready for later processing [106]. Later, in quiet moments and especially during sleep, the hippocampus briefly reactivates recent experiences in short bursts, as if skimming highlights [45, 76, 163]. These gentle replays draw out matching patterns in the cortex and, with repetition, encourage the same information to be set down again at a calmer tempo, where it settles beside what is already known and becomes easier to use [45, 76].

This repeated back-and-forth lets the fast hippocampal code and index guide stable updates on the cortical shelves [76, 106]. Vivid episode-specific details can be preserved when useful, while overlapping elements are gradually smoothed into more general, flexible knowledge. The outcome is a memory system that combines rapid capture with slow refinement, able both to hold onto life's distinct episodes and to extract durable patterns that support prediction, reasoning, and creative recombination.

*2.1.4 Implications for AI Memory Design.* Drawing on the cross-fertilization of cognitive psychology and neuroscience, we derive three actionable architectural patterns for agent memory that are designed to systematically address core challenges in agent memory functionality, including breaking context window constraints, realizing structured and generalizable long-term memory, and enabling efficient information integration for decision-making. These patterns are as follows:

- **Index and content separation pattern**: enabling efficient retrieval beyond context window limits. This pattern maintains compact episodic keys paired with a retriever that links to detailed, rich content in long-term storage—effectively overcoming the inherent capacity limitation of context windows. By doing so, it empowers agents to flexibly reinstate and synthesize past experiences on demand. For example, MemoryOS [73] implements a hierarchical memory architecture akin to an operating system, leveraging a lightweight index to dynamically swap data blocks from external storage into the limited active context.
- **Multiphase consolidation pattern**: fostering structured and generalizable long-term memory. This pattern transforms recent episodic traces into summaries, reflections, and reusable skills at strategic moments, thereby organizing long-term memory into a structured and generalizable form. Such a design enhances the utility of long-term memory for future decision-making [131]. Generative Agents [116] periodically synthesizes "reflections" from episodic traces, and Voyager [158] converts successful behaviors into a growing skill library.
- **Structured coordination pattern**: facilitating parallel maintenance and dynamic integration for decision-making. This pattern organizes the active workspace into specialized buffers overseen by a central controller, enabling the parallel, interference-free maintenance of verbal, visual, and tool-related outputs as well as their dynamic integration to support decision-making. For example, M3 Agent [99] explicitly structures an agent around a controller that schedules and coordinates distinct modules for perception, memory, and action, enabling parallel maintenance of heterogeneous information and deliberate integration before committing to a step.

### 2.2 Boundaries of "Memory" in AI

This section delineates the conceptual boundaries of AI memory from a multidimensional perspective. As shown in **Fig. 2**, we differentiate dynamic memory from static knowledge, contrast persistent storage with the ephemeral context window, and trace the cognitive progression from raw records to synthesized experience, clarifying the distinct conceptual contours of memory within AI systems.

*2.2.1 AI Memory vs. Agent Memory vs. LLM Memory.* We explicitly distinguish these terms by their respective emphases: LLM memory primarily emphasizes the computational kernel, Agent memory focuses on the functional workflow, and AI memory broadly represents the overarching cognitive concept bridging theory and application.

- **LLM memory** constitutes the low-level mechanics of prediction. It exists in two specific states: parametric memory embedded within pre-trained model weights and runtime memory managed via the context window. As noted in the discussion on Memory v.s. Context, this layer operates as the fundamental computational substrate. It prioritizes the accuracy of immediate generation within a bounded window over the maintenance of coherent autonomous behavior.
- **Agent memory** extends this foundation into a functional workflow that systematically supports autonomous behavior. Instead of producing isolated text, it coordinates the perception, planning, and action cycle so the system can decompose and execute complex tasks. By structuring data into distinct categories such as procedural, declarative, and metacognitive formats, Agent memory enables a system to learn from history. This layer facilitates the transition from static records to dynamic "experience" by enabling reflection and strategy refinement, which allows the agent to evolve its behavior based on past outcomes.
- **AI memory** represents the broadest definition of information persistence and evolution. It encompasses both the biological inspiration for artificial cognition and the ultimate goal of lifelong learning. While LLM memory provides the predictive engine and Agent memory manages task-oriented execution, AI memory defines the framework for lifelong evolution and experience accumulation to ensure a continuous, adaptive, and human-aligned lifecycle across diverse environments and long-term interactions.

*2.2.2 Memory vs. Knowledge.* Memory and knowledge play distinct roles in an AI agent. Memory acts as dynamic storage evolving through interaction, comprising both parametric weights and non-parametric external stores like vector databases or logs [53, 84, 114]. Characterized by timestamps and context, memory undergoes operations such as encoding and updating to capture recency and personal relevance [114, 116]. Conversely, knowledge represents static sedimentation consolidated for stability and reuse, existing as curated schemas, ontologies, and broad generalizations [53]. Unlike memory, which addresses what just happened, knowledge focuses on durable facts and abstractions. It prioritizes accuracy and consistency over immediacy to explain what typically works.

The boundary is permeable. Validated memories can be distilled into knowledge through consolidation, summarization, and schema alignment [53, 116], while knowledge conversely guides memory formation by shaping attention and prioritization. This interplay dictates that memory necessitates policies for retention, decay, and context-keyed retrieval [71, 114], while knowledge requires governance and provenance tracking [43]. Together, they allow agents to balance adaptability with coherence, personalization and stable reasoning across tasks.

*2.2.3 Memory vs. Context.* Context primarily represents the immediate execution environment within LLMs. It acts as a bounded computational buffer that holds the active tokens, system prompts, and temporary variables required for a specific inference step. From this perspective, context is strictly a runtime resource limited by the model's

architectural window size. Conversely, memory functions as an application-level state manager. It exists outside the model's ephemeral execution cycle to encapsulate the broader scope of user interactions and agent history. While context is cleared, shifted, or overwritten during processing, memory persists at the system interface level to maintain the continuity of the user relationship independent of any single inference call.

Despite the distinction between runtime execution and persistent state, the two elements form a vital operational loop. The context window serves as the processing unit where the model discrete memory entries from input data. These outputs are then appropriately integrated into memory storage. When a new task arises, the model retrieves relevant history from memory and injects it back into the context window to ground its reasoning. Recent research validates this bidirectional mechanism, as exemplified by MemoryOS [73], which introduces a hierarchical management interface that dynamically schedules long-term memory into the active context to support continuous self-evolution.

*2.2.4 Memory vs. Experience.* Memory and experience represent distinct levels of cognitive abstraction. Memory functions as the foundational record of specific interactions, preserving raw data points such as dialogue logs or sensory inputs as a static repository of "what happened." It provides high-fidelity retrieval but lacks inherent generalized utility. In contrast, experience constitutes a higher-order cognitive construct where these raw traces are synthesized into abstract, transferable patterns. Unlike a specific memory trace, experience encapsulates learned heuristics and causal models that enable an agent to generalize lessons from past contexts to novel tasks.

These two layers are bridged by a transformation process where experience acts as the functional product of processing memory. Through mechanisms like reflection and consolidation, the agent distills raw episodic records into refined cognitive strategies. Simultaneously, this distilled experience can be stored in memory, effectively treating high-level wisdom as a retrievable asset. This cycle ensures that the agent evolves beyond simple data recall, converting historical inputs into an expanding repertoire of adaptive capabilities.

## 3 Taxonomy of AI Memory

To provide a comprehensive understanding of the rapidly growing literature, We establish a **4W Memory Taxonomy** (i.e., **When-What-hoW-Which**) to systematically categorize AI memory systems, with each dimension anchored in a core interrogative word that addresses a fundamental characteristic of memory. **Table 1** systematically maps representative works under the 4W Memory Taxonomy.

- **W**hen (Lifecycle Dimension-§3.1): ***When*** *does memory exist and how long does it persist?* This dimension examines the temporal span of memory, ranging from transient input buffers to persistent cross-session storage. It addresses the memory duration in AI agent systems.
- **W**hat (Type Dimension-§3.2): ***What*** *type of information does memory capture?* Drawing from cognitive science, this dimension categorizes memory by the nature of knowledge it contains, including procedural skills, declarative facts, metacognitive reflections, and personalized models.
- Ho**W** (Storage Dimension-§3.3): ***HoW*** *is memory represented and stored?* This dimension explores the technical implementation of memory, from implicit parametric storage within model weights to explicit external representations, including text, vectors, and structured graphs.
- **W**hich (Modality Dimension-§3.4): ***Which*** *information formats does memory process?* This dimension classifies memory by information modalities. Single-modal memory exclusively processes textual data (e.g., literature abstracts, dialogue histories). Multimodal memory integrates text with heterogeneous formats such as images, audio, and video (e.g., image-text paired memory, audio-text synchronized memory, video-text fused memory).

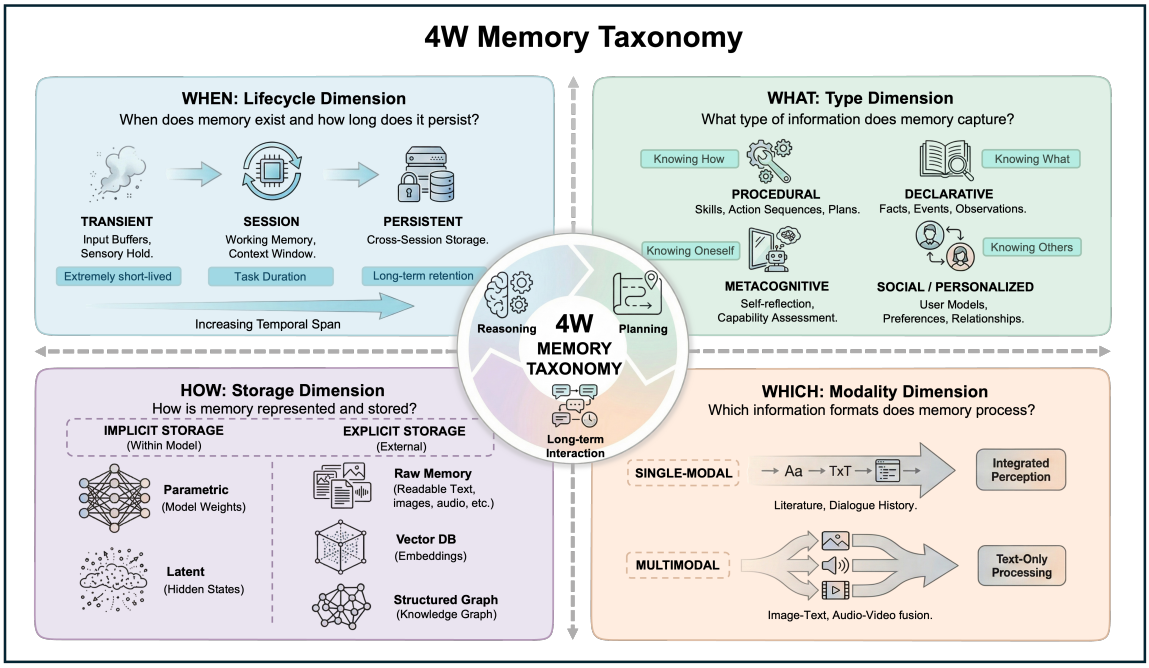


Fig. 3. 4W Memory Taxonomy for Systematic Classification of AI Memory Systems. It is structured around four orthogonal dimensions, corresponding to core interrogatives: When: Memory Lifecycle (temporal existence of memory); What: Memory Type (nature of stored information); How: Memory Storage (underlying storage mechanisms); Which: Memory Modality (specific memory modalities).

### 3.1 Classification by Memory Lifecycle

The lifecycle dimension examines the temporal span of memory in AI agent systems, focusing on how long memory persists and within what scope it remains accessible. This approach is inspired by cognitive psychology's Atkinson-Shiffrin model [5], which divides human memory into sensory, short-term, and long-term stages. In this taxonomy, we classify the dynamic nature of memory by describing a trajectory from transient input buffers, through session-bound working states, to persistent cross-session storage, enabling AI systems to address complex tasks such as multi-step planning and prolonged user interactions.

*3.1.1 Transient Memory.* Transient memory refers to extremely short-lived memory that exists only during immediate input processing, serving as a temporary buffer for perceptual inputs prior to further processing or storage. Analogous to human sensory memory in the Atkinson-Shiffrin model, it captures visual, auditory, or textual data streams without long-term retention. In AI agents, this typically manifests as input buffers that hold immediate environmental signals during preprocessing, which are often discarded after initial encoding. This memory type is characterized by high volatility and minimal persistence, which means information exists only momentarily in the processing pipeline [35]. For instance, in the Voyager framework [146], transient memory processes pixel inputs from Minecraft environments in real time, facilitating rapid responses to dynamic changes before relevant information is transferred to more persistent storage. Similarly, in transformer-based large language models, the key-value cache (KV Cache) serves as transient memory by temporarily storing key and value vectors to accelerate autoregressive generation, as optimized in layer-condensed approaches [165] and depth-wise compression techniques [94].

Table 1. Categorization of Representative Memory Research Works under the 4W Memory Taxonomy. T, S, and P are the abbreviations for transient memory, session memory, and persistent memory in the dimension of memory duration respectively.

| Work | Lifecycle | Storage | Content Type | OpenSource |
|---|---|---|---|---|
| ***I. Single-modal Memory*** | | | | |
| AMem [171] | Session | Text | Declarative | ✓ |
| AriGraph [3] | Persistent | Graph | Procedural | ✓ |
| AutoGPT [46] | Session | Text | Procedural | ✓ |
| Generative Agents [116] | T, S, P | Text | Procedural, Declarative, Personalized | ✓ |
| M+ [154] | Persistent | Latent, Vector | Procedural, Personalized | ✓ |
| Mem0 [24] | Persistent | Text, Vector, Graph | Procedural, Metacognitive, Personalized | ✓ |
| Memento [198] | Persistent | Text | Metacognitive | ✓ |
| MemGPT [114] | S, P | Parametric, Text, Vector | Procedural | ✓ |
| MemOS [90] | S, P | Text, Vector | Procedural | ✗ |
| MemoRAG [120] | Session | Latent | Declarative | ✓ |
| MemoryBank [197] | Persistent | Text | Procedural, Personalized | ✓ |
| MemoryLLM [153] | Session | Latent | Declarative | ✓ |
| MemoryOS [73] | S, P | Text, Vector | Declarative, Personalized | ✓ |
| Position [117] | Persistent | Text | Procedural | ✗ |
| ReAct [176] | Session | Text | Procedural, Declarative | ✓ |
| Reflexion [131] | Session | Text | Metacognitive | ✓ |
| Voyager [146] | T, S | Parametric, Text | Procedural, Declarative | ✓ |
| Zep [124] | Persistent | Graph | Personalized | ✓ |
| ***II. Multimodal Memory*** | | | | |
| AUGUSTUS [65] | T, P | Graph | Declarative | ✗ |
| Ego-R1 [143] | S, P | Text | Declarative | ✓ |
| Ego-LLaVA [129] | S, P | Text, Vector | Declarative | ✗ |
| EgoButler [174] | S, P | Text | Declarative | ✓ |
| Emboddied VideoAgent [36] | S, P | Text, Vector | Declarative | ✓ |
| HippoMM [93] | Persistent | Vector | Declarative | ✓ |
| JARVIS-1 [157] | S, P | Text, Vector | Procedural, Declarative | ✓ |
| M3-Agent [99] | S, P | Text, Graph | Declarative,Personalized | ✓ |
| MA-LMM [51] | Persistent | Vector | Declarative | ✓ |
| MIRIX [152] | Persistent | Text | Declarative | ✓ |
| MM-VID [92] | Persistent | Text | Declarative | ✓ |
| MovieChat [133] | S, P | Vector | Declarative | ✓ |
| Optimus [91] | S, P | Vector | Procedural, Declarative | ✓ |
| PENSIEVE [67] | Persistent | Text | Declarative | ✗ |
| VideoAgent [37] | S, P | Text, Vector | Declarative | ✓ |
| VideoLucy [202] | Session | Text | Declarative | ✓ |
| WorldMM [177] | S, P | Text, Vector, Graph | Declarative, Personalized | ✓ |

*3.1.2 Session Memory.* Session memory refers to memory that persists throughout a single task execution or conversational interaction but does not survive beyond that session. This form closely parallels the short-term or working memory component in the Atkinson-Shiffrin model and Working Memory model respectively, where information is held consciously for immediate use. It constitutes the core of an agent's active task processing, maintaining temporary state and context during ongoing operations. In LLM-based agents, this is commonly implemented through context windows. The key characteristic distinguishing session memory from transient memory is its deliberate retention during task execution; unlike transient buffers that immediately discard information, session memory actively maintains and manipulates information throughout the interaction lifecycle. For example, in-context learning approaches typically

leverage session memory to store instructions and contextual information that remain relevant during task completion [46, 97, 116, 131]. Recent studies on agent memory management explicitly formalize the concepts of session-bound working memory [73, 197]. These systems store and manage such information as a distinct component, aiming to enhance the model's capability in processing and retaining recent information throughout the interaction. By formalizing session memory as a standalone module rather than an implicit byproduct of the context window, these systems align more closely with human cognitive architectures while providing explicit lifecycle management.

*3.1.3 Persistent Memory.* Persistent memory refers to memory that survives beyond individual sessions and can be accessed across multiple interactions, tasks, or even different agent instantiations. In line with the long-term memory stage of the Atkinson-Shiffrin model, persistent memory enables lasting retention and retrieval of information across extended periods. It is stored in durable places like external databases, file systems, or model parameters, so it can be kept and reused over time. Unlike transient or session memory, it is not tied to a specific runtime context. This memory type is typically employed for storing knowledge that may not be immediately needed in the current interaction but remains relevant for future reasoning and decision-making. Persistent memory can be further categorized into parametric and non-parametric forms. Parametric persistent memory, embedded within model weights through pre-training or fine-tuning, is often employed in domain-specific tasks [12, 16, 23, 58, 62, 79, 98, 134, 159, 160, 186], where it supports rapid inference without explicit retrieval. In contrast, non-parametric persistent memory primarily relies on external memory banks with explicit storage mechanisms. Many approaches utilize file-based storage, vector databases, or structured repositories for preserving memories across sessions [21, 24, 27, 39, 68, 75, 82, 85, 115, 150, 153, 154, 156, 169, 197]. This externalization enables the agent to actively retrieve, utilize, and update its memory through interaction without relying solely on model parameters. Furthermore, persistent memory is generally reusable and transferable, allowing knowledge to be applied across different contexts or deployment scenarios.

### 3.2 Classification by Memory Type

The content type dimension examines the nature of knowledge that memory captures—what semantic category of information is being stored and for what purpose. Drawing from cognitive science's distinction between procedural and declarative memory [136], we extend this framework to encompass the unique requirements of AI agents, including metacognitive self-awareness and social modeling. This classification focuses on the functional role that different knowledge types play in agent behavior: procedural memory stores executable skills and action sequences (knowing how), declarative memory maintains factual and perceptual knowledge (knowing what), metacognitive memory supports self-reflection and capability assessment (knowing oneself), and social memory models other agents and users (knowing others). These content types are not merely organizational categories but reflect fundamentally different ways that memory contributes to agent intelligence, ranging from direct action execution to environmental understanding, self-improvement, and personalized interaction.

*3.2.1 Procedural Memory.* Procedural memory encapsulates executable knowledge, including skills, action workflows, and tool utilization. For AI agents, this means keeping action plans, learned behaviors, and action sequences. Its primary objective is the facilitation of goal-oriented actions rather than the storage of static facts. In the MemGPT framework [114], procedural memory stores multi-step planning sequences through hierarchical memory layers, enabling efficient task decomposition and execution. MemoryBank [197] employs procedural memory to capture and refine long-term decision patterns based on interaction history. Mem0 [24] maintains task-specific procedural preferences, learning user-preferred action sequences over time. DC [140] exemplifies procedural memory through its skill library,

where executable code snippets encode learned behaviors for Minecraft navigation and manipulation. Generative Agents [116] utilize procedural memory to store social interaction patterns and behavioral routines. Procedural memory improves efficiency by storing and reusing action sequences, leading to better task performance over time. However, it can be too specialized, may not adapt well to new situations, and can fail if flexibility is needed.

*3.2.2 Declarative Memory.* Declarative memory stores factual knowledge and perceptual observations—knowledge about what exists, what happened, and what was perceived. This corresponds to cognitive science's declarative (or explicit) memory system, encompassing both episodic memory (specific experiences) and semantic memory (general facts) [136]. In AI agents, declarative memory typically includes environmental observations, factual knowledge bases, event logs, and contextual descriptions. Generative Agents [116] employ memory streams to store perceptual observations and experienced events. ReAct [176] utilizes declarative memory in its think-observation-act loop, storing environmental observations and tool interaction results in context. VideoAgent [37] and similar multimodal systems rely heavily on declarative memory to maintain representations of visual environments and temporal event sequences. Declarative memory helps agents understand their environment, use context, and flexibly store facts. But it can be difficult to manage large amounts of data, decide what to keep, and keep information up-to-date as things change.

*3.2.3 Metacognitive Memory.* Metacognitive memory is knowledge about the agent's own thinking and abilities. It enables agents to track their performance, recognize strengths and weaknesses, reflect on past actions, and adjust strategies. This self-focused memory helps agents learn from experience, improve themselves, and become more robust and autonomous. For example, Reflexion [131] forms reflective language memory by recording task feedback (such as summaries of successes and failures, model-generated logs, etc.), allowing agents to analyze reasons for failures and adjust subsequent strategies, ultimately achieving self-reinforcement learning. The Memento framework leverages the powerful inductive and summarization abilities of large language models to automatically analyze agent behaviors and decision-making processes, extract suggestions for improvement, and archive them in self-cognitive memory, thereby continually optimizing strategy and improving performance [198]. In addition, LightSearcher [81] constructs a self-cognitive experience memory module by summarizing the agent's rollout, error experiences, assisting agents in self-examination and capability enhancement. These mechanisms are important for agents to handle complex tasks, adapt to changing situations, and adjust their actions based on both the environment and their own abilities.

*3.2.4 Personalized Memory.* Personalized Memory stores information about other agents and users, such as their preferences, behaviors, and relationships. This allows AI agents to remember and model individuals, track how relationships change, and personalize their interactions. For example, Mem0 maintains user preferences and identity information in production environments to enable fine-grained personalized services [24]; MemoryBank continuously adjusts its long-term memory structure based on user behavior to adapt to evolving user profiles [197]. Additionally, in MemoryOS, the user-related memory system supports efficient recording and retrieval of user inputs, maintains a user profile, and ensures the consistency and contextual relevance of multi-turn dialogues [73]. In practical applications, it is essential to integrate technical and regulatory means to balance the agent's personalized service capabilities with user data privacy.

### 3.3 Classification by Memory Storage

The storage dimension examines how memory is represented and stored in AI agent systems. It determines memory's physical form, access patterns, and computational properties. We categorize memory storage into two paradigms:

*3.3.1 Implicit Storage.* Implicit storage stores memory within the model architecture, either in its trained weights (parametric memory) or in hidden states during processing (latent memory). These representations are typically coupled with model inference, making them difficult to inspect or modify directly.

- **Parametric Memory**. Parametric memory refers to memory embedded directly within model parameters and weights, acquired through training processes such as pre-training [15], fine-tuning [195], or parameter-efficient adaptation methods like LoRA [10, 56]. This approach enables rapid inference without explicit retrieval. For example, Toolformer [128] uses finetuned model weights to store tool usage patterns and tool-related knowledge. Baijia [11] trains the model with role-specific dialogue data, internalizes Chinese ancient character role-playing abilities into its parametric memory. The primary advantage of parametric memory is its capacity for rapid, retrieval-free access, but it also suffers from catastrophic forgetting when learning new knowledge [101], expensive update costs requiring gradient-based optimization, and limited interpretability of stored knowledge. These limitations make parametric memory most suitable for relatively static foundational knowledge rather than rapidly evolving information.
- **Latent Memory**. Latent memory refers to memory that is not explicitly stored within model parameters but is implicitly represented through the model's learned latent space. This approach allows for the capture of complex, high-dimensional representations of knowledge that cannot be easily captured by traditional parametric storage methods. MemoRAG [120] uses an LLM to produce compact hidden-state memories, which captures global semantic structure. MemoryLLM [153] embeds a set of memory tokens within the model's latent space, enabling efficient memory management. M+ [154] further extends this idea into a cross-layer long-term memory architecture. Latent memory is particularly useful for representing abstract concepts, relationships, and patterns that are not easily represented by simple numerical vectors. In practice, it is often used in combination with parametric storage to capture both low-level and high-level features of the input data.

*3.3.2 Explicit Storage.* Explicit storage stores memory outside the model in formats such as text, vectors, or graphs. Compared to implicit storage, explicit forms make memory easier to retrieve, update, and interpret, but may be slower to access. Each method has its strengths and weaknesses regarding speed, storage, interpretability, and flexibility.

- **Raw Memory**. Raw memory stores information in a textual, visual, and auditory formats, including conversation histories, compressed dialogues, and other representations. It represents the most interpretable storage approach, preserving complete semantic content without information loss from compression or transformation. This makes text memory the most common form in LLM-based agents, as it naturally aligns with language models' input-output interfaces [168, 188]. In practice, numerous systems employ textual storage: MemoryOS [73] uses file-based textual storage for personal conversations. AMem [171] saves textual recollections for episodic memory. Mem0 [24] maintains long-term memory in text format for production agents. The primary advantages of raw text storage include maximal interpretability, seamless integration with LLM context windows, and preservation of nuanced semantic information, including tone, style, and implicit context.
- **Vector Memory**. Vector memory stores information as dense, continuous-valued vectors in high-dimensional semantic spaces, typically produced by neural storage models. It compresses semantic content into fixed-size numerical representations that enable efficient similarity-based retrieval through techniques such as approximate nearest neighbor search (e.g., FAISS [31]). Vector memory frequently operates in conjunction with raw text—systems use vector embeddings to index and retrieve textual content, exemplified by retrieval-augmented generation (RAG) architectures. It offers a good balance between text interpretability and efficient

storage. It is especially useful for capturing and retrieving information based on semantic similarity, rather than exact text matches. In practice, vector storage is widely implemented across agent memory systems: MemOS [90] employs vectorized memory within a unified management framework. M+ [154] utilizes latent-space vector representations for scalable long-term retention. Mem0 [24] maintains vector-encoded memory for production agents.

- **Graph Memory**. Structured graph memory encodes information as explicit networks of entities and their relationships, typically implemented using graph databases such as Neo4j [108]. Graph memory captures both knowledge items and their relationships, making it ideal for representing complex connections like social links or hierarchies. It is especially useful for tasks requiring multi-step or relational reasoning. Zep employs an atemporal knowledge graph architecture for agent memory management [124]. Mem0 integrates Neo4j as a memory backend for graph-based storage in production AI agents [24]. Cognee optimizes the interface between knowledge graphs and LLMs specifically for multi-hop question answering [105]. Graph-structured memory explicitly encodes relationships between entities, making it well-suited for relational reasoning and multi-hop queries over stored knowledge. While its structure supports interpretability, constructing and maintaining large-scale graphs can be computationally demanding. This form of memory is particularly effective in scenarios where relational structure is central, such as in social memory or knowledge-intensive tasks.

### 3.4 Classification by Modality Type

Classification by modality type categorizes AI memory according to the information formats it processes. From this perspective, memory systems are divided into single-modal and multimodal memory, with the classification focusing on whether memory mechanisms function with a single information modality or integrate multiple modalities to support perception, reasoning, long-horizon decision-making, and applications.

*3.4.1 Single-modal memory.* Single-modal memory focuses on storing, updating, and retrieving information from a single modality, with text being the most mature and widely adopted form due to its central role in LLM-based agents. Owing to its compact representation, single-modal text memory offers high computational efficiency and enables longer effective memory horizons under fixed resource constraints [48]. To overcome the limited context window of large language models, most systems introduce structured memory architectures rather than directly appending raw dialogue history. MemoryOS [73] and MemOS [90] draw inspiration from operating system memory management, organizing textual memory into multiple tiers with differentiated update, retention, and eviction strategies. Cognitive-inspired frameworks, such as those proposed in [3, 117, 162], partition memory into episodic, semantic, and procedural components to enhance selectivity and retrieval efficiency. Other systems focus on content abstraction and structural representation: Mem0 [24] employs summarization and selective updating pipelines to manage long-term dialogue memory, while Zep [124] leverages graph-based structures to model entities and relations within textual interactions.

*3.4.2 Multimodal memory.* Multimodal memory integrates information from multiple modalities, including text, images, audio, and video, enabling agents to perceive and reason about complex environments. This capability is typically supported by multimodal foundation models trained with contrastive learning or large-scale image–text alignment [122], which provide a unified semantic representation space across modalities. Existing multimodal memory systems fall into two major paradigms in terms of multimodal memory representation methods.

- **Raw Modality Representation**: It encodes raw multimodal data into high-dimensional embedding vectors, enabling fast access and feature reuse at the cost of increased storage and computation [65, 67]. For example, image-oriented memory systems such as Memory-QA [67] proposes a recall-oriented question-answering task together with the Pensieve system, which records visual, location, and temporal memories and retrieves relevant entries via multi-signal indexing for real-world memory augmentation. A large body of work within this paradigm focuses on video-centric memory [51, 59, 93, 143, 177]: Moviechat [133] adopts embedding-based memory to support long-video understanding and multimodal reasoning; VideoAgent [37] further constructs object-centric memory to facilitate entity tracking; VideoLucy [202] enables adaptive multimodal retrieval across extended temporal scales. For ultra-long egocentric applications, EgoLife [174] records daily-life streams to provide personal assistance over massive first-person video data. Regarding embodied settings and open-domain task execution, Optimus [91] and JARVIS-1 [157] demonstrate the effectiveness of embedding-based multimodal memory in long-horizon decision-making and multi-task execution in Minecraft environments, while Embodied VideoAgent [36] generalizes object-centric multimodal memory to real-world physical settings through egocentric perception.
- **Socratic Representation Paradigm**: It leverages a multimodal-to-text abstraction strategy within the Socratic paradigm [179], converting heterogeneous multimodal inputs into structured textual descriptions. By using language as a unified cross-modal intermediary, these methods significantly reduce storage overhead and improve interpretability, though they rely on accurate multimodal-to-text alignment. Ego-LLaVA [129] encodes egocentric visual experiences into compact textual memories, while MIRIX [152] and MM-VID [92] generate descriptive narratives from visual streams to support long-term retrieval and reasoning. M3-Agent [99] further extends this paradigm by constructing text-based episodic and semantic memories from real-time multimodal inputs to enable multi-turn reasoning over long video contexts.

## 4 AI Memory in Single-Agent System

This section provides a systematic overview of AI memory architectures and functional mechanisms designed for single-agent systems. To comprehensively illustrate how individual agents acquire memory capabilities, we organize the discussion along four key dimensions, with a classification of AI memory functions and objectives detailed in **Table 2**.

- **Memory Architecture (§4.1)**. We systematically reviews four dominant architectural paradigms in single-agent memory systems—hierarchical, OS-like, cognitive-evolutionary, and graph/temporal architectures—analyzing their design differences in information organization and task support.
- **Basic Functions (§4.2)**. We focus on the three foundational capabilities, i.e., storage, retrieval, and updating, and explore their technical implementations and functional roles in enabling agent behavior.
- **Advanced Capabilities (§4.3)**. We highlight self-evolution and multimodal association as key advanced functions, illustrating the transition from passive recall to active intelligence augmentation.
- **Paradigm Limitations (§4.4)**. We summarizes the structural limitations in information sharing, collaborative evolution, and collective intelligence, revealing the inherent constraints of single-agent memory paradigms in multi-agent contexts.

Table 2. Classification of AI Memory Functions and Objectives: Basic Function (Storage, Retrieval, Updating) & Advanced Function (Evolution, Association).

| Type | Function | Core Mechanisms & Strategies | Definition & Goal |
|---|---|---|---|
| **Basic** | Storage | • Procedural: Action plans & tool-calling sequences<br>• Declarative: Semantic facts & environmental logs<br>• Metacognitive: Thought processes & strategy feedback<br>• Personlized: User profiles & sInterests labels | Transforms observations into structured records with temporal, provenance, and personalized tags. |
| | Retrieval | • Vector-based: Semantic distance calculation<br>• Hierarchical: Abstract-to-detail routing<br>• Graph-based: Subgraph exploration & community search<br>• Multimodal: Cross-modal cues & episodic recall | Selects relevant memory slices to guide generation and reasoning. |
| | Updating | • Incremental: Non-lossy growth & latent integration<br>• Corrective: Weight regulation & side-memory isolation<br>• Consolidation: Recursive summarization & heat-based promotion<br>• Forgetting: Active unlearning & inverted labels | Rectifies errors and synchronizes content with new observations. |
| **Advanced** | Evolution | • Pattern Distillation: Reasoning trajectory extraction<br>• Skill Libraries: Executable workflow induction<br>• Strategy Optimization: Theory of Mind & dynamic belief systems | Distills AI experiences into adaptive, logic-preserving skills and policies. |
| | Association | • Spatiotemporal alignment: Entity/timestamp/location mapping<br>• Fusion: Early fusion & cross-attention grounding<br>• Linking: Graph-style records & provenance tracking | Integrates fragmented multimodal signals into coherent situational models. |

### 4.1 Typical Agent Memory Architecture

This section focuses on the external memory systems of agents and categorizes three key memory architectures primarily from the perspective of information organization forms. Specifically, hierarchical architecture organizes memory via hierarchical storage structures with dynamic management; cognitive and evolutionary architectures mimic human cognition or adopt the Theory of Mind; graph and temporal architectures organize information through graph structures (e.g., entity-relation graphs) or temporal models to capture complex relationships.

*4.1.1 Hierarchical Memory Architecture.* Hierarchical Memory Architecture [49, 52, 57, 73, 111, 137, 145, 178, 187, 200] is specifically designed for agent systems to resolve the inherent contradiction between the limited context window capacity of LLMs and the insatiable demand for long-term information storage and retrieval. Drawing inspiration from cognitive psychology models of human memory (e.g., the Atkinson–Shiffrin model [5]), many of these architectures adopt a layered design that mimics the hierarchical organization of human memory. For instance, HMT [52] stratifies long contextual information into hierarchical sensory, short, and long-term layers, enabling efficient hierarchical management and adaptive retrieval of information at different abstraction levels. H-MEM [137] structures memory into a four-level hierarchy based on the degree of semantic abstraction, complemented by an index-based routing mechanism to enable layer-wise retrieval. This design effectively avoids exhaustive similarity computations, significantly improving retrieval efficiency for large-scale memory.

*4.1.2 OS-like Memory Architectures.* Drawing inspiration from operating system design, these architectures employ hierarchical storage and dynamic management mechanisms to address challenges in memory coherence and resource allocation during long-term interactions [73, 90, 107, 114]. MemGPT [114] emulates the paging techniques of traditional

operating systems to transfer data between the limited context window and external storage. MemoryOS [73] adopts an OS-inspired hierarchical architecture comprising short-term, mid-term, and long-term storage units, utilizing a 'heat-based' segmented paging strategy to govern the dynamic migration and eviction of dialogue context. MEMOS [90] treats heterogeneous knowledge as a manageable system resource and unifies plaintext, activation, and parameter-level memories through a "MemCube" unit to enable controllable scheduling and long-term cognitive evolution.

*4.1.3 Cognitive Evolution Memory Architectures.* Cognitive-driven memory approaches either simulate human cognitive processes or incorporate Theory of Mind [118], which is defined in the AI field as the ability to model and reason about the mental states and intentions of humans. This capability enables agents to develop evolvable memory and strategy systems for self-optimization [1, 65, 69, 72, 112, 185]. For example, AUGUSTUS [65] mimics human cognitive processes by establishing a four-stage closed loop of "Encode-Store-Retrieve-Act" and building a graph-structured multimodal context memory to handle multimedia interactions. Nemori [112] organizes conversational streams into structured episodic narratives and proactively evolves its semantic knowledge base through a "predict-calibrate" cycle that distills insights from discrepancies between internal forecasts and actual interactions.

*4.1.4 Graph and Temporal Memory Architectures.* Targeting the enhancement of agent memory, graph-based memory approaches leverage graph structures—particularly knowledge graphs—to model complex relational dependencies, encode temporal dynamics for precise memory lifecycle management, and improve reasoning accuracy [22, 24, 49, 50, 69, 83, 124, 126, 127]. For instance, Zep [124] introduces a temporal knowledge graph architecture that preserves both non-lossy raw data and abstracted facts through layered episodic and semantic subgraphs, employing a dual-timeline mechanism for fine-grained lifecycle control. Mem0 [24] constructs a graph-structured memory of entities and relationships to capture intricate dependencies among dialogue elements, while MemTree [126] dynamically organizes conversational histories into tree-based hierarchical structures, maintaining branching logic and long-term coherence. Collectively, these graph-based memory systems substantially enhance multi-hop and temporal reasoning capabilities, while markedly reducing retrieval latency compared with traditional RAG-based memory architectures.

### 4.2 Basic Functions of AI Memory

AI memory turns observations and interactions into durable, actionable states that support planning, learning, and safety. This section introduces basic functions that include memory encoding, storage, retrieval, updating, and higher-order functions that enable self-evolution and cross-modal association. Memory functions grant an agent explicit control over the transformation of memory into a usable state, dictating precisely what information is captured, when it is retrieved, how it is refined, and when it is discarded.

*4.2.1 Memory Storage.* Memory storage serves as the core module of AI agents, primarily functioning to transform fragmented observational data into structured and durable memory records while establishing a standardized indexing architecture to underpin subsequent retrieval. To achieve precise access, each memory unit is configured with functional components, including timestamps for temporal anchoring, source identifiers for data provenance tracking, and structured semantic fields (e.g., entities, relations, task tags) as fundamental indexing dimensions; these metadata ensure memory traceability in complex tasks. In terms of content-type classification, the stored content falls into four categories. Procedural memory encompasses action plans, learned behaviors, and tool-calling sequences to optimize task execution efficiency. Declarative memory involves factual knowledge and perceptual observations, building the agent's world cognition foundation via environmental logs and semantic facts. Metacognitive memory records the

agent's own thought processes and feedback summaries to support self-performance tracking and strategy adjustment, enabling failure reflection and behavioral evolution. Personalized memory maintains user profiles to ensure interaction consistency and personalized experiences. From the perspective of storage format, memory is divided into two types: explicit storage (e.g., textual logs, knowledge graphs, task records) refers to directly accessible and interpretable content; implicit storage corresponds to information encoded in model parameters (e.g., continuous learning weight updates, historical interaction embeddings), which is not human-readable but can be effectively utilized during agent inference. Detailed discussions are provided in **Section §3.2** and **Section §3.3**.

*4.2.2 Memory Retrieval.* The core of memory retrieval research lies in the precise retrieval and integration of information from large-scale memory repositories to guide the generation process, thereby alleviating hallucinations and enhancing reasoning capabilities. These approaches are broadly classified into four primary architectures: **vector-based retrieval, graph-based retrieval, hierarchical retrieval, and multimodal retrieval**.

Vector-based memory retrieval [22, 84] maps discrete memory content into embedding vector spaces to break lexical matching constraints and capture latent semantic correlations. For queries like "animals that catch mice," it converts queries into vectors and calculates semantic distances, enabling retrieval of "feline habits" documents without explicit keyword overlap [84]. Integrating semantic similarity with task schemas ensures retrieved content fits context limits, boosting precision and mitigating hallucinations. Hierarchical memory retrieval [73, 120] structures memory into semantic abstraction tiers (summaries to details) to resolve context window limitations and noise interference. Adopting a "directory-first" logic, it locates macro intents first then drills down for details. Graph-based retrieval frameworks [24, 124] simulate the associative memory mechanism of the human brain by representing memory elements as interconnected nodes and edges, effectively overcoming the limitations of traditional vector retrieval in handling complex logical reasoning and long-range dependencies. For example, Zep [124] implements this through its Graphiti engine, which combines breadth-first search with hierarchical community summarization to achieve global memory retrieval that spans temporal dimensions and incorporates temporal validity filtering. Multimodal memory retrieval [93, 133] integrates visual information with semantic labels, extending the scope of retrieval tasks beyond the textual domain. HippoMM [93] achieves this by structuring continuous audiovisual streams into long-term episodic representations, enabling the system to recall complete episodic memories from only partial or cross-modal cues. Similarly, MovieChat [133] manages memory by condensing dense observations into persistent sparse records, empowering it to either retrieve information across the entire video history or pinpoint specific moments preserved in its active memory buffer.

*4.2.3 Memory Updating.* Memory updating is the process of revising, replacing, or consolidating existing stored content, ensuring that an agent prevents error recurrence by rectifying erroneous or outdated information and integrating fragmented knowledge as new data arrives. We categorize single-agent memory updates into four types: **incremental updates, corrective updates, consolidation updates, and forgetting updates**.

Incremental memory updates focus on continuously injecting newly perceived experiences and information into the memory bank without interfering with existing knowledge. For instance, Zep [124] dynamically synthesizes information in a non-lossy and incremental manner into a temporally-aware knowledge graph. Similarly, MemoryLLM [153] and its extension M+ [154] achieve incremental expansion by integrating new information into the model's latent space and dynamically offloading excess memory tokens to an external storage. However, such continuous accumulation poses the challenge of information inflation, where unchecked growth of redundant or noisy data can significantly degrade retrieval efficiency and increase computational overhead. Corrective memory updates aim to rectify outdated

or erroneous knowledge within the model or adjust its perception of specific facts. For example, H-MEM [137] achieves real-time correction and adaptive updating of memory strength by combining traditional forgetting curves with a dynamic weight regulation mechanism based on user feedback. WISE [149] adopts a dual-parametric memory scheme to physically isolate corrections in a side memory from pre-trained knowledge in the main memory. Consolidation memory updates optimize storage structures and enhance retrieval efficiency through the semantic abstraction and summarization of fragmented memories. For instance, in MemoryOS [73], when the comprehensive Heat score of a specific Mid-Term Memory segment exceeds a predefined threshold, the system invokes an LLM to extract evolving user traits and factual knowledge. This Heat score is calculated based on factors including access frequency, interaction depth, and time decay. The extracted information is subsequently updated into the Long-term Personal Memory to ensure persistent persona consistency. LightMem [38] employs a cognition-inspired sleep-time consolidation mechanism to reorganize, de-duplicate, and abstract long-term memory entries during offline periods. MemoryField [4] treats memory as dynamic nodes in a gravitational field, employing a fusion mechanism to merge semantically similar and spatially close nodes into higher-density representations for semantic abstraction and redundancy reduction. Finally, memory forgetting updates maintain memory system efficiency by actively deleting or inhibiting redundant, sensitive, or low-value information, where MEOW [47] utilizes "inverted fact" labels for stealthy unlearning through fine-tuning.

### 4.3 Advanced Functions of AI Memory

Going far beyond the basic functions, AI memory not only preserves information but also processes it to transform raw data into actionable intelligence for downstream tasks. This subsection addresses self-evolution, which distills iteratively optimized patterns from AI experience, and association, which integrates fragmented multimodal signals. These advanced functions collectively synthesize isolated observations into a robust understanding of complex environments.

*4.3.1 Self-Evolution.* Evolution refers to an agent's ability to dynamically iterate and optimize acquired knowledge, skills, and behavioral strategies over continuous interaction and task execution [72, 139]. Instead of accumulating static surface details, the agent organizes and refines AI experience into evolvable structures (e.g., adaptive goals, adjustable constraints, updated causal relations, iterative action schemas). For new tasks, these evolutionary frameworks guide the adaptation of existing knowledge, reduce the cost of incremental learning, and distill raw episodic traces into concise, logic-preserving yet adaptable skills, policies, or prompts. Leveraging such refined and evolvable experience, the agent evolves its functional capabilities to adapt to different users, domains, and tools with fewer demonstrations and less retraining, while enhancing robustness to noise, novelty, and distribution shifts. Recent studies validate the role of structured AI experience in knowledge and skill evolution: LightSearcher [81] boosts DeepSearch by distilling successful reasoning and tool-invocation trajectories into experiential memory, enabling reusable proven patterns that evolve to balance accuracy and efficiency across dynamic task demands. Voyager [146] similarly builds an expanding executable skill library that evolves through continuous task practice to solve unseen tasks in new Minecraft worlds, accelerating mastery and mitigating forgetting. Collectively, these systems demonstrate that structuring AI experience as evolvable skills/workflows anchored to source episodes enables adaptive iteration and functional enhancement under new constraints with minimal fine-tuning.

*4.3.2 Association.* Association refers to integrating multimodal signals (text, vision, audio, interactions) into coherent situational models for memory construction. In practice, fusion aligns entities, timestamps, and locations to map related cues (e.g., face, spoken name, caption) to a single memory node. Practical pipelines combine early fusion for low-level cues, cross-attention/retrieval modules for mid-level grounding, and graph-style linking for durable

memory structures—reducing ambiguity, improving reference resolution, and preserving memory continuity across frames/dialogue turns. Summaries and embeddings keep fused content compact for short-term memory, while linked, provenanced records ensure long-term memory traceability. M3-Agent [99] exemplifies this memory integration in multimodal video (speech+chat): it aligns speaker faces, names, slide timestamps, and referenced entities to build an action/fact memory graph. Early fusion stabilizes low-level cues (face tracks, speaker diarization), cross-attention grounds verbal mentions to visual regions, and the agent summarizes segments into embeddings for short-term memory reasoning while storing linked records for long-term memory reference. Mem-0g [24], which is a graph-based Mem0 extension, preserves entities/relations alongside embeddings to enrich structured memory, retrieving via vector search + graph traversal. It delivers modest gains over the base system on long-dialogue benchmarks, with its fused, traceable memory structure supporting follow-up reasoning and cross-project task planning.

### 4.4 Limitations of Single-Agent Memory Paradigms

Existing agent memory architectures are primarily tailored for single-agent scenarios. However, these architectures lack sufficient adaptability to memory management in multi-agent systems, and direct extension of such frameworks often results in inefficiencies and coordination bottlenecks [18, 43]. For example, explicit memory, implemented via retrieval-augmented generation (RAG) in personal assistant applications, is characterized by private contextual storage spaces. This inherent design inadvertently perpetuates information silos in multi-agent systems and culminates in a critical loss of cross-role information. Conversely, implicit memory focuses on internalizing experience via parameter updates, which typically involves continuous fine-tuning whose prohibitive computational overhead makes it expensive for multi-agent synchronization. Under the mechanism of single-agent memory, agents attempting to collaborate via these isolated memory stacks incur structural failures, with the manifestations of such failures falling into the following key aspects:

- **Memory Misalignment**. Silent failures emerge when agents' perceptions of the global state diverge. An agent may optimize its outputs based on obsolete or fragmented private memories, unaware that the project's trajectory has been altered by peers. Consequently, while individual components may be locally valid, they often prove globally incompatible, thereby compromising the coherence of the overall system.
- **The Redundancy Cycle**. In the absence of a unified progress record, agents inevitably engage in duplicative labor. For instance, one agent may expend computational resources addressing a sub-problem that has already been resolved and archived in the private memory of another agent. This engenders a recursive cycle of redundant computation and storage inefficiency, resulting in excessive token consumption and suboptimal resource utilization.
- **Stagnation of Collective Intelligence**. Memory isolation inhibits the system's capacity for adaptive learning. Valuable insights, such as specific API workarounds, remain siloed within individual agents. This prevents the accumulation of shared knowledge, forcing subsequent agents to rediscover solutions de novo rather than leveraging a foundational knowledge base, thus stifling the evolution of collective intelligence.

## 5 Memory Mechanisms in Multi-Agent Systems (MAS)

We explore the foundational architecture of collective intelligence in Multi-Agent Systems (MAS), aiming to bridge the gap between transient interaction and persistent knowledge. Addressing the limitations of isolated memory models, which inevitably lead to redundancy and information silos. We organize the discussion on MAS memory mechanisms into two core components (i.e., Communication Mechanisms and Sharing Mechanisms), where the framework is detailed in **Fig.** 4, and we synthesize relevant existing MAS works in **Table** 3.

- **Communica tion Mechanisms (§5.1)**: We define the spectrum of communication modalities, distinguishing between *explicit communication* (e.g., natural language, structured schemas) that ensures interpretability and *implicit communication* (e.g., latent representation) designed for high-speed, state-based coordination.
- **Memory Sharing Mechanisms (§5.2)**: We propose a comprehensive taxonomy of memory sharing categorized by *task-level* and *step-level* granularities, addressing the distinct targets of knowledge management and heterogeneous optimization priorities required for effective multi-agent collaboration.

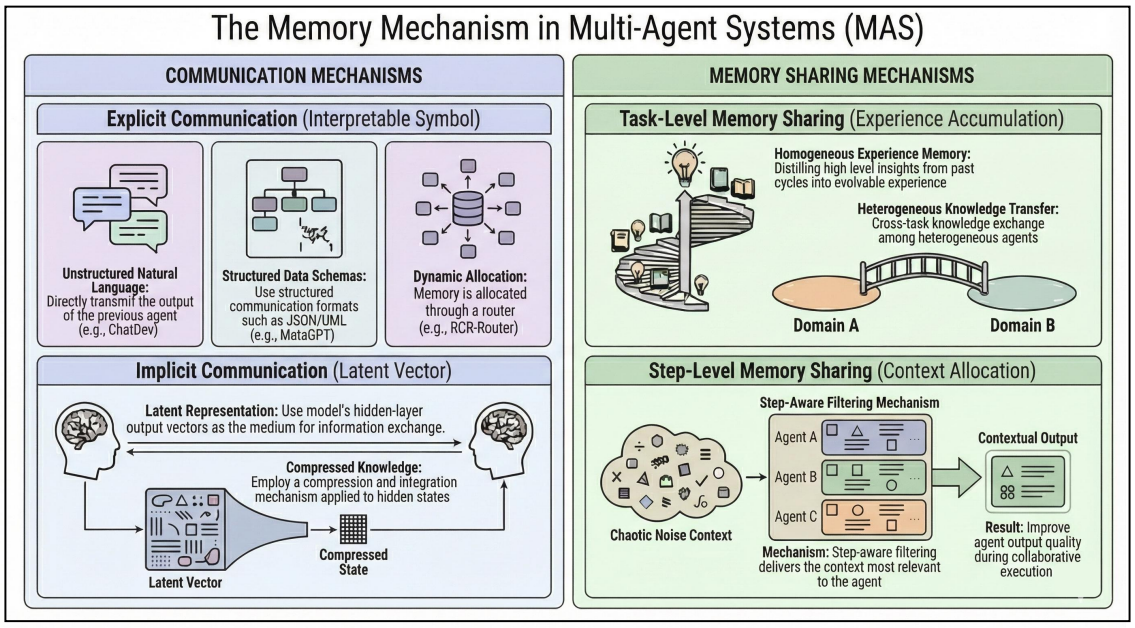


Fig. 4. The Framework Categorizes Memory Mechanisms in MAS into Two Dimensions: Communication Mechanisms (left), which serve as the foundation and are divided into explicit (e.g., natural language, data schemas) and implicit (e.g., latent vector) types; Memory Sharing Mechanisms (right), which are implemented via sharing type and divided into Task-Level and Step-Level sharing.

### 5.1 Communication Mechanisms in MAS

Effective collaboration within multi-agent systems hinges on communication mediated by memory sharing. Communication in MAS exhibits two primary modalities that underpin the foundational framework of multi-agent communication: explicit, symbolic exchange and implicit, state-based coordination. This section explores memory-centric communication mechanisms in MAS and architectures of shared memory critical to collaborative task execution.

Table 3. Taxonomy of Memory Mechanisms in Existing Multi-Agent System (MAS) Works: Memory Communication Mechanisms (Explicit/Implicit) and Memory Sharing Modes (Task-Level/Step-Level).

| MAS System | Memory Mechanism | Description of MAS Framework | Link |
|---|---|---|---|
| | | ***I. Explicit Communication*** | |
| **ChatDEV** [119] | Unstructured Natural Language | Uses "role-based prompting" where prompts include context summaries, embedding a volatile form of memory directly into the template. | |
| **MetaGPT** [54] | Structured Data Schemas | Enforces "Structured Communication Interfaces" (e.g., UML), ensuring outputs are machine-interpretable and reliably consumable. | |
| | | ***II. Implicit Communication*** | |
| **LatentMAS** [201] | Latent Representation | Enables agents to communicate entirely in continuous latent space, sharing hidden embeddings as "latent thoughts" for lossless exchange. | |
| **Dense Comm.** [167] | Dense Communication | Proposes bypassing the decoding process to enable dense information flow, moving reasoning out of the text domain. | |
| **Thought Comm.** [196] | Thought-to-Thought | Facilitates "thought-to-thought" interaction where internal states represent reasoning chains akin to telepathic collaboration. | |
| **InterLat** [34] | Compressed Knowledge | Applies compression to latent representations to optimize inference efficiency while preserving rich internal information. | |
| **Activation Comm.** [123] | Activation Processing | Demonstrates that processing continuous states (activations) significantly reduces computational overhead compared to text decoding. | |
| | | ***III. Task-Level Memory Sharing*** | |
| **MemoryOS++**[1] | Swarm Experience | Searches swarm experience through systematic human experience accumulation to enable high-quality responses. | |
| **G-Memory** [181] | Memory Abstraction | Hierarchically structures memory to distill high-level insights from past failures/successes for longitudinal evolution. | |
| **SEDM** [170] | Evolutionary Distillation | Explicitly distills reasoning trajectories into evolvable wisdom, allowing agents to adapt over time. | |
| **MS** [42] | Lateral Knowledge Transfer | Establishes shared knowledge pools for cross-domain transfer, enabling agents to replicate proven solution paths in similar contexts. | |
| | | ***IV. Step-Level Memory Sharing*** | |
| **RCR-Router** [95] | Role-Aware Context Routing | Dynamically routes information based on functional roles to solve the "noise-context trade-off" and reduce cognitive load. | |

*5.1.1 Explicit Communication.* Explicit communication involves the deliberate transmission of symbolic information between agents. In LLM-based systems, this is a complex spectrum, ranging from unstructured natural language dialogue to highly structured, formalized protocols designed to eliminate ambiguity and ensure interoperability.

- **Unstructured Natural Language**. A communication paradigm in which agents coordinate through natural-language interactions guided by explicitly assigned roles. For instance, ChatDEV employs role-based prompts that include not only its assigned role (e.g., "You are a senior software engineer"), but also a summary of relevant context from prior interactions [119]. This embeds a simple, volatile form of memory directly into the prompt template. This approach, while flexible, suffers from the inherent ambiguity and chattiness of natural language, often leading to misunderstandings, off-topic loops, and a high degree of token consumption.
- **Structured Data Schemas**. A communication mechanism that constrain inter-agent information exchange to predefined, machine-interpretable formats. For example, MetaGPT moves beyond chat and mandates that agents communicate using a predefined "schema and format," often based on UML (Unified Modeling Language) diagrams. This structural protocol ensures that the outputs of one agent (e.g., an architect's system design, formatted as a class diagram) are directly and reliably consumable by the next agent in the workflow [54]. This structured data, often serialized in formats like JSON or YAML, represents a deliberate and high-fidelity method for transferring memory fragments (i.e., task-relevant knowledge). This paradigm extends to standard operating procedures, where agents in a workflow do not just communicate but commit structured artifacts to a shared workspace, which triggers the next agent's operations.

- **Dynamic Allocation**. Dynamic Allocation entails that information exchange no longer relies on static one-to-one interactions between agents, but is instead dynamically routed from a shared memory space to relevant agents based on task requirements. In this paradigm, agents write intermediate results, partial reasoning traces, or task-relevant knowledge into a common memory repository, utilizing formats such as natural language or structured data schemas mentioned in the previous communication methods. The RCR-Router is responsible for retrieving, filtering, and redistributing this information to appropriate agents as needed. By decoupling information production from consumption, this approach replaces fixed communication topologies with a memory-centric many-to-many information flow, thereby achieving more flexible and scalable agent collaboration [95].

*5.1.2 Implicit Communication.* In contrast to explicit communication, which takes place between agents through direct, deliberate messaging. Implicit communication of MAS enables coordination without such inter-agent interaction, operating instead through internal processing within individual agent programs. In this mode, agents infer the intent or state of others by observing a shared environment or a shared representation of their internal states.

- **Latent Representation**. This paradigm involves multi-agent systems where agents directly share their internal continuous latent representations (hidden embeddings) instead of discrete natural language tokens. Pioneering works in this domain propose that bypassing the decoding process enables "dense communication" [167] or "thought-to-thought" interaction [196], allowing for pure latent collaboration with lossless information exchange. By moving reasoning out of the text domain, these methods aim to achieve higher expressive capacity akin to telepathic collaboration. LatentMAS [201] exemplifies this framework by enabling agents to communicate and coordinate entirely in a continuous latent space. In such systems, agents generate and share internal hidden embeddings as “latent thoughts” stored in a shared working memory.
- **Compressed Knowledge**. This approach functions as an advanced paradigm of latent communication, distinguished by the integration of compression mechanisms applied to the model’s final hidden states. The primary objective is to optimize inference efficiency while preserving the semantic fidelity of high-dimensional internal information. Research on activation-based communication [123] demonstrates that processing continuous states significantly mitigates the computational overhead inherent in decoding and re-encoding natural language. Building upon these efficiency principles, Interlat [34] introduces a framework wherein agents optionally compress latent representations to further accelerate inference. This methodology suggests that while downsampling complex internal states into discrete tokens limits reasoning abilities, directly transmitting latent states through efficient compression techniques enables agents to better utilize subtle internal information. Consequently, this strategy enables agents to achieve competitive performance and accelerated inference, outperforming traditional chain-of-thought prompting while fostering more exploratory behavior compared to single-agent baselines.

### 5.2 Memory Sharing Mechanism in MAS

In multi-agent systems, shared memory serves as the substrate for collective intelligence. Current research optimizes this mechanism at two distinct granularities: task-level memory sharing, which focuses on the retention and transfer of knowledge across different execution lifecycles, and step-level memory sharing, which optimizes the precise distribution of information within the granular workflow of a single collaborative task.

*5.2.1 Task-Level Memory Sharing.* Task-level memory sharing refers to the mechanism of consolidating experiences from distinct task executions to facilitate long-term evolution and cross-domain transfer. This approach redefines memory not as a mere execution buffer, but as a persistent vehicle, serving not only the immediate task but also as a profound reservoir for accumulated experience.

- **Homogeneous Experience Accumulation**. This memory-sharing mechanism refers to the process where an agent team accumulates experience throughout the execution of a specific task. The primary challenge here lies in transforming raw historical data into evolvable wisdom or experience. Instead of retaining linear execution logs, which lead to retrieval inefficiencies, effective methodologies employ memory abstraction to hierarchically structure memory. This process distills high-level insights, procedural skills, and abstract strategies from past failures and successes. By synthesizing these derived lessons, agents can continuously optimize their performance on subsequent tasks. For instance, recent works [170] demonstrate how explicit distillation of reasoning trajectories enables agents to self-evolve and adapt over time. Notably, MemoryOS++[1] serves as a seminal chatbot platform that leverages swarm experience search and utilization. It automatically extracts useful information from conversations to co-build a swarm experience repository, thereby enabling high-quality responses in the chatbot. This platform exemplifies the aforementioned approach by facilitating problem-solving through the systematic accumulation of human experience.
- **Heterogeneous Information Transfer**. It represents a memory-sharing mechanism that facilitates information exchange among distinct agents performing heterogeneous tasks. Even when agents operate in unrelated domains, they often encounter sub-problems with similar underlying logic such as planning or conflict resolution. It enables lateral knowledge transfer, establishing shared information pools where agents can retrieve and replicate proven solution paths from peers. By leveraging these shared experiences, agents can effectively bypass the need for zero-shot exploration in structurally similar contexts, thereby achieving rapid adaptation across different task boundaries [42].

*5.2.2 Step-Level Memory Sharing.* Step-level sharing refers to the process of dynamically allocating specific information to relevant agents during the granular execution phases of a single collaborative workflow. This mechanism addresses the "noise-context trade-off" inherent in multi-agent collaboration, where broadcasting global states rapidly exhausts context windows and dilutes attention. The core methodology here is context routing. Instead of maintaining a fully synchronized global state for all participants, this approach implements role-aware filtering mechanisms. The system analyzes the functional role of each agent and the current phase of the task to determine the strict necessity of information. By pruning irrelevant context and delivering only the critical information slices required for the immediate next step, the system reduces computational overhead and maintains agent focus. Frameworks utilizing such dynamic routing strategies, such as RCR-Router [95], exemplify how optimizing intra-task information flow significantly enhances collaborative efficiency without overwhelming the agents' cognitive capacity.

[1] https://baijia.online/share/

## 6 Evaluation on AI Memory

We present a systematic overview of the evaluation landscape for LLM-based agent memory. Notably, there remains a lack of unified assessment criteria, which is a critical gap that motivates the structured synthesis below.

- **Evaluation Metrics (§6.1)**: We propose a comprehensive taxonomy comprising four core categories: memory retrieval capability, dynamic updating capability, advanced cognitive capability, and system efficiency.
- **Representative Benchmarks (§6.2)**: We review representative benchmarks that provide the essential resources, datasets, and protocols for standardized testing.
- **Evaluation Challenges (§6.3)**: We critically analyze three primary challenges: the obstacles of dataset construction, ambiguity in performance attribution, and the dilemma of evaluation metrics.

Table 4. Taxonomy of Evaluation Dimensions for LLM-based Agent Memory: With Core Dimensions, Descriptions, and Metrics, spanning from Memory Retrieval Capability & Dynamic Updating Capability & Advanced Cognition Capability & System Efficiency.

| Sub-Dimension | Description | Representative Metrics |
|---|---|---|
| | ***I. Memory Retrieval Capability*** | |
| **Retrieval Performance** | Directly evaluates the quality of retrieved chunks regarding coverage, signal-to-noise ratio, and ranking order against ground-truth memory. | **Recall@$k$**, **Precision@$k$**, **NDCG@$k$** |
| **Response Correctness** | Indirectly assesses retrieval quality via the success rate of answering extractive questions based on the retrieved context. | **Accuracy**, **F1-Score**, **BLEU-N**, **ROUGE-L** |
| | ***II. Dynamic Updating Capability*** | |
| **Memory Modification** | Evaluates the system's ability to correctly overwrite old states with new conflicting information (e.g., user preference changes) while ignoring stale data. | **Update Accuracy**, **Hallucination Rate**, **Omission Rate** |
| **Memory Writing** | Verifies the completeness and factual correctness of the information encoded from raw interactions into the memory module without loss or hallucination. | **Recall**, **Accuracy**, **F1-score** |
| **Memory Forgetting** | Assessing the algorithmic erasure of specific data influence from parametric memory (machine unlearning) while preserving the integrity of unrelated knowledge. | **Truth Ratio**, **ROUGE-L Recall**, |
| | ***III. Advanced Cognitive Capability*** | |
| **Generalization** | The ability to transfer knowledge to unseen tasks via zero/few-shot learning, utilizing stored task trajectories or abstracting universal rules. | **Success Rate** |
| **Temporal Perception** | The capability to reconstruct event sequences and track state changes from unstructured text by resolving timestamps. | **Kendall's $\tau$**, **Accuracy** |
| **Personalization** | Measures the effectiveness of supporting user-centric interactions and improving satisfaction by persistently storing context information. | **Accuracy**, **Human or LLM Score** |
| | ***IV. System Efficiency*** | |
| **Latency** | The precise time cost for specific operations (e.g., Retrieval or Writing) excluding the network latency of LLM API generation. | **Percentile Latency** |
| **Token Overhead** | The total number of tokens consumed by packaging the retrieved memory into the prompt for a single interaction round. | **Tokens Consumed** |
| **Storage Efficiency** | It evaluates the memory module's capability to minimize physical storage footprint as much as possible while retaining all critical information from the raw text under limited constraints. | **Storage Cost** |

### 6.1 Evaluation Metrics for Memory Mechanisms

We establish a comprehensive taxonomy of evaluation dimensions for AI memory, organized into four primary categories: **memory retrieval capability**, **dynamic updating capability**, **advanced cognitive capability**, and **system efficiency**. A detailed taxonomy of these evaluation dimensions for AI memory is summarized in **Table 4**.

*6.1.1 Memory Retrieval Capability.* Memory retrieval serves as the fundamental interface between the agent and its stored history. The effectiveness of a memory system is primarily determined by its capacity to accurately and comprehensively locate information relevant to the current query. We categorize the evaluation of this capability into two distinct sub-dimensions: Retrieval Performance and Response Correctness.

- **Retrieval Performance**. Retrieval performance serves as a direct evaluation of the memory module's quality, focusing on whether the system can precisely fetch relevant segments from the database. This dimension integrates coverage, precision, and ranking quality. To quantify this, three representative metrics are widely used as follows:
  - **Recall@$k$**. This metric calculates the proportion of ground-truth relevant memories successfully retrieved in the top-$k$ results.
  - **Precision@$k$**. It measures the proportion of retrieved chunks that are actually relevant.
  - **NDCG@$k$**. Normalized Discounted Cumulative Gain (NDCG) evaluates ranking quality by assigning higher scores when relevant memories are positioned near the top of the context window, which is critical for mitigating the "Lost-in-the-Middle" phenomenon in LLMs.

- **Response Correctness**. Apart from direct retrieval performance, many studies evaluate memory retrieval capability indirectly through the success rate in downstream tasks, where the answers can be directly located in the raw text. Response correctness can be evaluated by:
  - **Accuracy**. This metric measures the ratio of questions correctly answered to the total number of queries.
  - **F1-Score**. It calculates the harmonic mean of precision and recall, balancing the exactness and completeness of the text overlap between the prediction and ground truth.
  - **BLEU-$N$** (Bilingual Evaluation Understudy). This metric focuses on precision by calculating the fraction of $N$-grams in the generated text that appear in the reference answer.
  - **ROUGE-L** (Recall-Oriented Understudy for Gisting Evaluation). It emphasizes recall by measuring the Longest Common Subsequence (LCS) between the candidate and the reference text, capturing sentence-level structure.

In terms of direct retrieval performance, Recall@$k$ is extensively employed to quantify coverage by comparing retrieved content against ground-truth evidence [63, 87, 189]. Distinct from this coverage-centric view, Precision@$k$ is applied in [77] to measure the proportion of relevant program trajectories within the retrieved results. Furthermore, addressing the order of information, NDCG@$k$ is applied to evaluate ranking quality, ensuring that critical memory segments are prioritized at the top of the list [164]. Complementing these direct measures, the assessment of Response Correctness functions as a proxy for retrieval quality, operating on the premise that accurate downstream answers imply the successful retrieval of relevant context. This indirect methodology is utilized in several studies [74, 114, 180, 189], where the memory module's effectiveness is validated by the agent's performance in answering factual questions derived directly from the raw text.

*6.1.2 Dynamic Updating Capability.* Beyond static information retrieval, the ability to evolve memory dynamically is crucial for long-term interaction. This dimension evaluates whether the memory system can correctly maintain the freshness of its knowledge base. We examine this capability through three sub-dimensions: Memory Modification, Memory Writing and Memory Forgetting.

- **Memory Modification**. Memory modification assesses the system's capacity to correctly modify existing records when new, conflicting information is presented (e.g., a change in user preference). The core challenge lies in prioritizing the latest state over obsolete data.To evaluate this dimension, the following metrics are commonly employed:
  - **Update Accuracy**. This metric is calculated as the ratio of correctly modified or replaced memories to the total number of target modifications.
  - **Hallucination Rate**. It measures the proportion of modifications that introduce factual errors or logical conflicts.
  - **Omission Rate**. This metric quantifies the frequency with which the system ignores necessary modifications, leaving the old memory unchanged.
- **Memory Writing**. It evaluates the fidelity and completeness of the process where raw interaction text is converted into stored memory. It aims to ensure that no critical information is lost or hallucinated during the saving phase. To evaluate this dimension, the following key metrics are commonly used:
  - **Memory Recall**. This metric evaluates the completeness of the storage process by calculating the ratio of key facts successfully committed to memory to the total number of gold standard facts.
  - **Memory Accuracy**. It assesses the correctness of the written entries, verifying whether the stored content is factually consistent with the dialog history without fabrication.
  - **F1-score**. This metric provides a balanced score to evaluate the overall quality of the memory writing process. It is calculated as the harmonic mean of memory recall and memory accuracy, thereby simultaneously assessing both the completeness and the fidelity of the stored memories.
- **Memory Forgetting**. Memory forgetting, often formalized as machine unlearning, refers to the algorithmic process of selectively erasing the influence of specific data from the model's parametric memory while preserving the integrity of unrelated knowledge [121]. The motivation for this capability is driven by critical requirements, including privacy compliance, copyright protection for unlicensed works, and safety alignment to remove hazardous capabilities encoded during pre-training. To evaluate this dimension, the following primary metrics are commonly employed:
  - **Truth Ratio**. It calculates the normalized likelihood of the model generating the correct (deleted) answer versus a perturbed alternative. It serves as a statistical gold standard to determine if the unlearned model is indistinguishable from a model retrained from scratch.
  - **ROUGE-L Recall**. It measures the textual overlap between the model's response and the ground truth of the deleted target. Lower scores indicate effective unlearning, verifying the model's inability to recall specific memory traces under various probes.

For memory modification, evaluation methods in this category typically verify either the modification process or the final state. Regarding the modification process verification, HaluMem [20] adopts a direct inspection approach, utilizing APIs to access internal memory states and verify execution correctness. It explicitly evaluates the reliability of memory modification through three key metrics: Accuracy, Hallucination Rate, and Omission Rate. In contrast,

outcome-oriented methods [66, 141] quantify the accuracy of the agent's behavior in reflecting the new state, employing either multiple-choice tasks to directly test awareness of updated information, or LLM judges to verify that generated responses align with the user's latest status [64, 164]. For memory forgetting, to rigorously verify if a memory is truly erased, TOFU [103] employs the Truth Ratio metric to determine if the unlearned model exhibits no statistically significant difference in behavior from a gold-standard model that never encoded that specific memory. Targeting the elimination of hazardous memory, WMDP [88] evaluates the reduction in accuracy on dangerous domains to random chance levels, utilizing linear probes to verify that latent memory traces are unrecoverable from internal activations. To rigorously evaluate the efficacy of memory forgetting under adversarial conditions, RWKU [70] utilizes the ROUGE-L recall score to measure the extent to which the model retains specific target memories across diverse probing and attack scenarios. Finally, evaluating the forgetting granularity across different memory forms, MUSE [130] assesses the decoupling of verbatim and knowledge memory, employing membership inference attacks to verify that no privacy leakage remains compared to a model trained without the target data.

*6.1.3 Advanced Cognitive Capability.* Advanced cognitive capability represents the agent's ability to transcend simple storage and retrieval, utilizing memory for higher-order reasoning. We analyze this capability through three key aspects: Generalization, Temporal Perception, and Personalization.

- **Generalization**. Memory generalization refers to the capability of large language models to effectively transfer and apply acquired knowledge or skills to unseen tasks. By utilizing complete task trajectories or abstracting universal templates stored in memory, LLM-based agents can exhibit superior performance in untrained scenarios. Evaluating generalization performance serves as a direct indicator of the memory module's effectiveness in supporting knowledge transfer.To evaluate this dimension, researchers primarily measure the **success rate** of the agent in some unseen tasks based on its stored memory.
- **Temporal Perception**. Temporal perception refers to the agent's capability to maintain and update a coherent timeline of events and entity states across interactions, enabling consistent reasoning over evolving temporal dependencies. By identifying explicit timestamps or resolving implicit relative references (e.g., "last week") into specific dates, the memory module empowers agents with robust temporal awareness.To evaluate this dimension, the following metrics are typically used:
  - **Kendall's $\tau$**. This coefficient is used to measure the rank correlation between the predicted event order and the ground-truth timeline.
  - **Accuracy**. It calculates the percentage of success for the agent in executing temporal perception tasks.
- **Personalization**. Personalization measures the agent's ability to utilize long-term memory to provide tailored services based on user history, identity, and behavioral patterns.To evaluate this dimension, the following metrics are typically used:
  - **Accuracy**. Used for objective tasks, this metric measures the percentage of responses that match the user's specific profile.
  - **Human or LLM-based Scoring**. This approach is employed to assess subjective aspects of the interaction, such as coherence and emotional resonance.

For memory generalization, this dimension is predominantly evaluated across cross-task, cross-domain, and cross-website datasets [158, 194]. In addition, an approach assesses the agent's memory generalization to induce rules from dialogue streams and classify unseen inputs, utilizing accuracy as the primary metric [60]. Regarding temporal

perception, evaluation methodologies center on two core tasks: timeline reconstruction and state tracking. In timeline reconstruction tasks, large language models are tasked with reordering events, with performance measured against ground-truth timelines via Kendall's $\tau$ coefficient computed on timestamped entity events [64]. For state tracking, temporal reasoning is assessed by measuring the accuracy in tracing user preference trajectories [66] or inferring entity states under date constraints [74]. The assessment of personalization is typically divided into objective alignment and subjective quality. Regarding objective alignment, system performance is evaluated via the accuracy of a multiple-choice task designed to identify the user's latest specific preferences [66]. Conversely, for subjective quality, researchers rely on human or LLM-based scoring to capture the "soft" aspects of interaction. Specifically, human annotators are utilized to assess naturalness and coherence of responses from large language models [87, 96, 197], while GPT-4 is employed as a judge to specifically score the emotional resonance of generated responses [100].

*6.1.4 System Efficiency.* System efficiency evaluates the engineering viability for real-world deployment, serving as a critical determinant for the scalability and user experience of memory-augmented agents. While retrieval accuracy is paramount, the practical adoption of these systems depends heavily on their operational responsiveness and economic feasibility. Accordingly, this dimension focuses on the operational overhead introduced by the memory module, covering Latency, Token Overhead, and Storage Efficiency.

- **Latency**. Latency evaluation measures the operational overhead of memory modules, primarily encompassing retrieval and update latencies. Specifically, retrieval latency quantifies the time required to locate and fetch relevant information from the memory store, and update latency accounts for the duration of modifying the memory state. By isolating these system-level costs, we can precisely evaluate the responsiveness of various memory architectures. To evaluate this dimension, the primary metric is **Percentile Latency**, which describes the performance distribution by indicating the maximum time taken by a specific percentage of requests.
- **Token Overhead**. Token overhead evaluates the input cost associated with maintaining long-term memory. It quantifies the number of tokens consumed by packaging the retrieved memory into the context window for a single interaction round. To evaluate this dimension, researchers typically measure **Tokens Consumed**, calculated by counting the average number of tokens in the prompts constructed by the memory module, where a lower count implies better cost-efficiency.
- **Storage Efficiency**. The memory storage efficiency of an LLM agent refers to its ability to store all essential information from the original text using as little storage space as possible under limited memory constraints. The metric for this dimension is **Storage Cost**, which is used to measure the physical storage size occupied by the processed memory structure.

Evaluation of latency is typically stratified into retrieval and update phases. For retrieval latency, the focus is on the time required to filter and fetch relevant information, which is measured using automated code profiling to record the precise duration of the retrieval function's execution [20, 24, 96, 171]. For update latency, it measures the interval from when new information is written until it becomes retrievable. While this is often recorded directly via writing function timestamps [20, 63, 141, 189], an alternative method known as "Query Probing" is introduced in [24]. This approach involves injecting a specific fact and periodically querying the system, defining latency as the elapsed time until the correct answer is provided. Regarding token overhead, existing works quantify this by calculating the average token count of the prompts constructed by the memory module, serving as a direct proxy for the system's economic efficiency [73, 124, 171]. To quantify this, researchers [100] contrast the storage occupancy of a baseline 'append-only' strategy with an optimized compression mechanism." Notably, a comparative analysis is conducted in [24] quantifying

the storage requirements relative to the raw conversation context (avg. ~26k tokens). They find that while Mem0 achieves significant compression (~7k tokens) by encoding dialogues into concise natural language representations, graph-based approaches vary drastically: optimized graphs (i.e., Mem0g) incur moderate overhead (~14k tokens), whereas redundant designs (i.e., Zep) can lead to massive inflation (> 600k tokens), exceeding the raw context by over 20 times—due to the design choice of caching full abstractive summaries at every node.

### 6.2 Evaluation Benchmarks

To systematically organize the benchmarks of memory evaluation, we propose a taxonomy that classifies representative benchmarks based on the core characteristics of their evaluation tasks. A detailed mapping of all benchmarks to this taxonomy is presented in **Table 5**:

- **Static Memory Evaluation**. Emphasizing memory retrieval from fixed, non-updating input, the temporal nature of task input is the fundamental distinction between Static and Dynamic Memory Evaluation.
- **Dynamic Memory Evaluation**. Focuses on assessing the agent's core ability to manage memory updates and adapt to evolving contextual information.
- **Personalization Memory Evaluation**. Evaluates the memory's key capacity to synthesize and maintain evolving user profiles and personalized preferences.
- **Environment Memory Evaluation**. Centers on assessing the memory's practical effectiveness in supporting sequential actions in complex external environments.
- **Multimodal Memory Evaluation**. Tests the memory's ability to align and retrieve spatio-temporal information across heterogeneous non-text modalities.

*6.2.1 Static Memory Evaluation.* Benchmarks in this category assess memory in a "static" capacity. The primary challenge stems from the difficulty of accurately retrieving relevant information from memory. **LoCoMo** [102] and **EpisodicGen** [64] focus on narrative grounding; the former leverages evolving temporal event graphs to synthesize conversations for assessing memory capabilities over simulated months, while the latter generates synthetic narratives to evaluate the retrieval of episodic memories via specific cues. In addition, LongBench[13] evaluates the fidelity of information retention and retrieval within massive bilingual context windows, while its successor LongBench v2[14] assesses the active utilization of retained memory for deep understanding and reasoning in various tasks. Moreover, **RULER** [55] evaluates the effective memory capacity of LLMs by measuring performance retention across increasingly long synthetic tasks that require information retrieval, multi-hop tracing, and aggregation. The **HotpotQA** dataset [175] can be adapted for memory evaluation by distributing supporting evidence across multiple interactions, requiring the agent to store and retrieve relevant information from memory to answer multi-hop questions.

*6.2.2 Dynamic Memory Evaluation.* Unlike static setups, these benchmarks evaluate memory over a temporal narrative where information evolves, gets updated, or is contradicted over time. The core challenge is not retrieval, but rather the dynamic management of memory. **MemoryAgentBench** [60] focuses on incremental interactions, evaluating an agent's capability to dynamically update its memory state under conflicting information. **MemoryBench** [2] evaluates memory updating in an on-policy learning setting, where an LLM system receives simulated user feedback and updates its memory accordingly. **HaluMem** [20] investigates reliability by detecting hallucinations during memory extraction and updating. **DialSim** [74] evaluates dynamic memory maintenance within streaming multi-party dialogues by injecting

Table 5. Taxonomy of Representative Benchmarks for LLM-based Agent Memory. Benchmarks are categorized by their primary evaluation objective. The **Evaluation Tasks** column describes the specific testing methodologies used, and **Key Metrics** lists the primary metrics employed.

| Benchmark | Description | Evaluation Tasks | Key Metrics | Link |
|---|---|---|---|---|
| | | ***I. Static Memory Evaluation*** | | |
| **LoCoMo** [102] | Simulates very long-term conversations grounded in evolving temporal event graphs to ensure causal consistency. | Long-Term QA, Event Graph Summarization, Dialogue Generation | F1 Score, FactScore, ROUGE, BLEU | |
| **EpisodicGen** [64] | Uses synthetic novels with controlled event metadata to test spatiotemporal recall. | Cue-based Recall, Entity State Tracking, Chronological Ordering | Optimistic F1, Kendall's $\tau$ | |
| **LongBench** [13] | Bilingual, multi-task benchmark for evaluating retrieval and reasoning on massive static contexts. | Multi-Document QA, Summarization, Key-Passage Retrieval, Code Completion | ROUGE-L, F1, Accuracy, Edit Similarity | |
| **LongBench v2** [14] | Evaluates deep reasoning on realistic long-context data using human-verified, retrieval-resistant questions. | Multi-Document QA, Code Repository Understanding, Long-Dialogue History | Accuracy, Human Expert Gap | |
| **RULER** [55] | Synthetic suite testing effective context limits with complex retrieval and tracing tasks. | Multi-key NIAH, Variable Tracking, Frequent Words Extraction | Effective Context Length, Weighted Avg. Accuracy | |
| **HotpotQA** [175] | Evaluates multi-hop reasoning by requiring agents to bridge information across multiple documents. | Multi-hop QA, Supporting Fact Extraction, Factoid Comparison | Exact Match, F1, Joint F1 | |
| | | ***II. Dynamic Memory Evaluation*** | | |
| **MemoryAgentBench** [60] | Simulates incremental memory usage by injecting text chunks sequentially to test updates. | Incremental Event QA, Counterfactual Fact Updating, Test-Time Classification | Accuracy, Recall@5, F1, Failure Rate | |
| **HaluMem** [20] | Evaluates hallucinations during memory extraction and updating in continuous dialogues. | Memory Extraction, Memory Updating, Dynamic QA (Conflict) | Integrity (Recall), Accuracy, FMR, Hallucination/Omission Rate | |
| **MemoryBench** [2] | Evaluates performance improvement over time by learning from a stream of user feedback. | Legal Judgment Generation, Research Synthesis, Feedback Adaptation | LLM-Judge Score, F1, ROUGE, Accuracy | |
| **DialSim** [74] | Simulates multi-year, multi-party social interactions to evaluate memory retention in a streaming setting. | Spontaneous QA, Temporal Reasoning, Unanswerable Question Detection | Accuracy | |
| **BEAM** [142] | Synthetic long conversations (up to 10M tokens) testing complex, dynamic memory abilities. | Contradiction Resolution, Event Ordering, Information Update | Nugget-based Score, Kendall's Tau-b | |
| | | ***III. Personalization Memory Evaluation*** | | |
| **MemBench** [141] | Evaluates retention of explicit facts and abstraction of implicit user preferences (reflective memory). | Reflective Memory Inference, Factual QA, Knowledge Updating | Accuracy, Recall, Capacity, Latency | |
| **MemSim** [189] | Simulates daily life by generating consistent user profiles to produce factual queries. | Single-hop QA, Comparative QA, Aggregative QA | Accuracy, Recall@5, Latency | |
| **PERSONAMEM** [66] | Simulates long-term interactions with evolving personas to evaluate dynamic user profiling. | Track Preference Evolution, Recall User Facts, Personalized Recommendation | Accuracy | |
| **LongMemEval** [164] | Tests retention and updating of evolving user profiles in long, scalable interactions. | Multi-session Reasoning, Knowledge Updates, Temporal Reasoning | Accuracy (LLM-Judge), Recall@k, NDCG@k | |
| **PerLTQA** [33] | Simulates personal character archives (profiles, social graphs, events) for personalized memory tests. | Memory Classification, Fact Retrieval, Response Synthesis (QA) | MAP, Recall@k, Accuracy, F1 | |
| **PREFEVAL** [192] | Evaluates personalization by embedding user preferences within long distractor conversations. | Preference-Adherent Generation, Multiple-Choice Classification | Accuracy, Violation Rate, Hallucination Rate | |
| | | ***IV. Agent Environment Memory Evaluation*** | | |
| **WebChoreArena** [109] | A web simulation environment for tedious 'chores' requiring massive cross-page reasoning. | Massive Memory Retrieval, Memory-based Calculation, State Tracking | Success Rate, Task Completion Rate | |
| **MT-Mind2Web** [30] | Simulates multi-turn conversational web navigation on HTML snapshots. | Contextual Action Prediction, HTML Element Selection, Generalization | Element Acc, Op. F1, SSR, TSR | |
| **StoryBench** [144] | Uses interactive fiction games to test navigation of branching storylines and error recovery. | Branching Decision Making, Causal Reasoning, Self-Recovery | Success Count, Accuracy, Retry Count | |
| | | ***V. Multimodal Memory Evaluation*** | | |
| **Video-MME** [40] | First comprehensive multi-modal benchmark for MLLMs on diverse short-to-long videos (up to 1h) with expert-annotated QA. | Perception QA, Cognition QA, Synopsis QA | Accuracy | |
| **MLVU** [199] | Multi-task benchmark for long video understanding (3min–2h) across diverse sources and multiple tasks. | Holistic Task, Single-Detail Task, Multi-Detail Task | Accuracy, GPT-4 Score | |
| **LVBench** [151] | Extreme long-video benchmark (avg. >1h, up to hours) for ultra-long context in real-world videos. | Entity Recognition, Event Understanding, Key Info Retrieval | F1, Accuracy, IoU | |
| **M3-Bench** [99] | Evaluates memory over continuous video/audio streams by building entity-centric memory graphs. | Cross-modal Reasoning, Person Understanding, Multi-evidence Reasoning | Accuracy, Identity F1 | |
| **EgoSchema** [104] | Diagnostic QA benchmark for very long-form first-person video understanding from Ego4D. | Multiple-Choice QA, Temporal Event Reasoning | Accuracy | |
| **EgoLifeQA** [174] | Long-context QA benchmark from week-long egocentric life videos for personalized daily assistance. | EventRecall, EntityLog, TaskMaster, HabitInsight, RelationMap | Accuracy | |
| **Memory-QA** [67] | Evaluates recall from a repository of ego-centric images aligned with spatio-temporal metadata. | Visual Recall, Multimodal QA, Spatio-Temporal Retrieval | Accuracy, Recall@k, nDCG | |
| **MMNeedle** [147] | Tests visual retrieval by hiding target sub-images within large, stitched image sequences. | Visual Needle Retrieval, Multi-Needle Localization, Negative Sample Detection | Existence Acc, Index Acc, Exact Acc | |

spontaneous questions that require agents to reason over an accumulating history. Finally, **BEAM** [142] challenges agents by evaluating their capacity to update memory as facts change throughout long, evolving conversations.

*6.2.3 Personalization Memory Evaluation.* This category focuses on assessing the efficacy of memory mechanisms in constructing and maintaining dynamic user profiles, where agent is expected to track evolving personas and infer implicit preferences over long-term interactions. **PERSONAMEM** [66] and **LongMemEval** [164] evaluate memory's capability in updating dynamic user information across extensive interactions. **PREFEVAL** [192] specifically tests the agent's ability to infer and adhere to implicit user preferences buried within long distractor conversations. **MemBench** [141] evaluates agents on their capacity to synthesize implicit user preferences alongside explicit factual details to construct dynamic, multi-level user profiles across interactive and observational scenarios. **PerLTQA** [33] evaluates personal long-term memory by testing an agent's ability to synthesize answers from a character's archive. **MemSim** [189] uses Bayesian networks to generate consistent user profiles for memory evaluation.

*6.2.4 Agent Environment Memory Evaluation.* Agent Environment Memory Evaluation focuses on assessing memory mechanisms under interactive, stateful environments, where agents must retain, update, and exploit past information across extended action sequences. It measures memory not through isolated recall, but through its sustained impact on sequential decision-making in environments such as web navigation and interactive games. **WebChoreArena** [109] is an environment-based benchmark that extends WebArena with labor-intensive web tasks, assessing agents' abilities to maintain long-term memory across multi-page interactions. **MT-Mind2Web** [30] evaluates an agent's memory module in a conversational web navigation task. It requires the agent to leverage a memory of the multi-turn dialogue history to interpret context-dependent instructions, such as those with anaphora and ellipsis, and ground them into concrete actions on a webpage. **StoryBench** [144] utilizes interactive fiction games to test causal reasoning and error recovery, where the agent's memory of past decisions directly impacts future states.

*6.2.5 Multimodal Memory Evaluation.* This category addresses memory over non-text modalities, including image, audio and video. **Video-MME** [40], **MLVU** [199], **LVBench** [151] and **M3-Bench** [99] evaluate memory capabilities across diverse real-world video types, focusing on multi-modal understanding, long temporal memory, and broad task coverage. **EgoSchema** [104] and **EgoLife** [174] target continuous first-person perspective videos, with a focus on fine-grained event tracking, long-term memory, habit inference, and real-world life understanding over extended timescales. **Memory-QA** [67] acts as a visual "second brain", tasking agents with recalling information from a repository of stored ego-centric images combined with spatio-temporal metadata. Lastly, **MMNeedle** [147] adapts the needle-in-a-haystack test to the visual domain, assessing an agent's ability to precisely locate sub-images within large visual contexts.

### 6.3 Evaluation Challenges

Based on our comprehensive analysis of existing benchmarks, we identify three practical obstacles that persist in the rigorous evaluation of AI memory.

*6.3.1 Obstacles of Dataset Construction.* The acquisition of authentic, long-term dialogue data is fraught with significant hurdles, particularly regarding severe privacy risks and prohibitive data collection costs. Consequently, researchers are frequently compelled to rely on alternative sources, such as synthetic data generation or crowdsourced datasets, as practical substitutes [87, 125]. However, these surrogates often fall short, failing to adequately capture the intricate nuances and complex dynamics inherent in genuine human interactions. Furthermore, the production of high-quality synthetic data is not trivial, and it often necessitates the utilization of state-of-the-art, closed-source foundation models

(e.g., GPT-4). This reliance inevitably imposes a substantial financial burden, thereby significantly escalating the overall costs associated with benchmark construction [102].

*6.3.2 Ambiguity in Performance Attribution.* A central challenge in evaluating agent memory lies in the difficulty of attributing downstream performance to the memory module in tasks where memory retrieval is deeply intertwined with multi-step reasoning or long-horizon planning. Most existing benchmarks adopt end-to-end evaluation paradigms, assessing agent memory solely through aggregated downstream task performance. However, such tasks exhibit heterogeneous memory dependency patterns. While some primarily require explicit information retrieval, others demand that retrieved memories be tightly integrated into complex reasoning or planning processes. In the latter case, task-level failures may stem from memory retrieval errors, reasoning deficiencies, or their intricate interaction, rendering performance attribution inherently ambiguous. Consequently, end-to-end evaluation introduces confounding factors that obscure error localization and hinder the identification of memory-specific failures, such as memory-induced hallucinations or omissions [20]. The field currently lacks benchmarks and evaluation protocols to explicitly model task-specific memory dependency or isolate memory module effectiveness from an agent's overall behavior [188].

*6.3.3 Dilemma of Evaluation Metrics.* Beyond the challenges in dataset construction and performance attribution, a third critical obstacle lies in the very methods employed to evaluate AI memory systems. First, the evaluation criteria adopted by existing benchmarks are highly fragmented, and there is currently no unified or comparable metric framework. Benchmarks differ substantially in their choice of metrics, and in cases where a benchmark focuses on a specific aspect, a model may achieve high scores through targeted optimization rather than genuinely improving the core capabilities of its memory mechanism. This phenomenon is analogous to the well-known issue in machine learning where optimizing for benchmark metrics can inadvertently degrade real-world task performance. Consequently, there remains a lack of a comprehensive benchmark capable of assessing the overall capabilities of AI memory [102, 113]. Besides, current evaluations of AI memory largely lack dynamic metrics, which refer to metrics that assess the evolutionary trajectory of memory states, and the development of suitable dynamic metrics remains largely unaddressed. Dynamic evaluation has long been applied in other domains: for instance, in clinical practice, clinicians track the dynamic changes of vital signs (e.g., blood pressure) instead of single measurements to assess health status. By contrast, existing memory benchmarks typically evaluate models at discrete time points, neglecting the dynamic trajectories of memory updates and forgetting over extended interactions.

## 7 AI Memory Applications

We present a taxonomy of memory-enabled AI applications across two paradigm: Single-Agent and Multi-Agent Memory Systems. Building on the scenario summaries in **Table 6**, this taxonomy clarifies the role of memory in AI systems across diverse domains:

- **Single-Agent Memory Applications (§7.1)**: It introduces applications where an individual AI agent is augmented with independent long-term memory. Such memory allows a single agent to maintain persistent context, remember user preferences, and accumulate experience over extended or disjoint interactions, enabling personalization and continuity beyond the fixed context window.
- **Multi-Agent Memory Applications (§7.2)**: We discuss multi-agent scenarios leveraging shared memory for collaboration and collective reasoning, where a common memory space enables coordinated task execution while introducing concurrency, provenance, and access control challenges.

Table 6. Representative Applications of AI Memory Systems: including Memory Descriptions, Key Technical Requirements, and Representative Systems, categorized into Single-agent and Multi-agent paradigms.

| Applications | Memory Discriptions | Key requirements and challenges | Representative systems |
|---|---|---|---|
| | | ***I. Single-agent*** | |
| Dialogue Assistants | Cross-session long-term user profile/preference memory. | • Selective retrieval under context limits<br>• Update/forget to avoid stale facts<br>• Mostly implicit for users; explicit for developers | ChatGPT Memory<br>Claude memory tool<br>MemGPT<br>LangChain Memory<br>Rasa (tracker store)<br>Baijia MemoryOS |
| Embodied Robotics | Environmental multimodal world-model memory (visual–spatial–action traces). | • Multimodal grounding (language, vision, spatial)<br>• Spatio-temporal indexing and fast updates<br>• Implicit interaction in end-user products | Voyager<br>Home-Robot |
| Personalized Recommendation | Long-term user interaction-history memory (with temporal dynamics). | • Preference drift and temporal dynamics<br>• Privacy-aware storage and access<br>• Traceable memory for interpretability | Amazon Personalize<br>Google Gemini |
| Medical Diagnosis | Longitudinal patient record memory (EHR timelines, medications, tests, notes). | • Safety-critical recall requirements<br>• Verifiable and auditable memory<br>• Explicit clinician queries and traceability | Doctor AI<br>Ada Health<br>Buoy Health |
| Content Generation | Narrative/persona / world-state memory for long-form consistency. | • Canon and constraint preservation<br>• Scalable summarization and retrieval<br>• Hybrid explicit-implicit interaction<br>• Role-playing and personalized alignment | AI Dungeon<br>NovelAI<br>SillyTavern<br>Character.AI<br>BaiJia Agents |
| | | ***II. Multi-agent*** | |
| Collaborative Workflows | Shared workspace memory for plans, artifacts, decisions, and tool outputs. | • Concurrency and version control<br>• Provenance and conflict handling<br>• Explicit agent read/write | MetaGPT<br>AutoGen<br>CrewAI<br>LangGraph<br>OpenDevin |
| Open-world Simulation | Event-log memory plus agent-centric episodic memory for long-term social coherence. | • Salience-based retention<br>• Private vs. shared memory separation<br>• Mostly implicit for observers | Generative Agents<br>AI Town |
| Scientific Discovery | Shared knowledge-base memory with structured provenance (facts, hypotheses, evidence). | • Schema constraints and precision<br>• Strong provenance for verification<br>• Human-inspectable memory | SciAgents<br>MRAgent<br>PaperQA |
| Financial Decision Support | Time-sensitive shared belief-and-evidence memory (signals, rationales, constraints). | • Time-aware decay and freshness<br>• Low-latency updates<br>• Traceability for audit | FinGPT<br>OpenBB |
| Experience Sharing Platform | Community experience/workflow memory for cross-agent reuse and attribution. | • Deduplication and quality control<br>• Attribution, privacy, and access control<br>• Retrieval for adaptation to new tasks<br>• Value Assessment and Personalized Alignment | BaiJia MemoryOS++ |

## 7.1 Single-Agent Memory Applications

Single-agent systems leverage independent, persistent memory components to overcome the fixed context window limitations of LLMs [168]. Long-term memory enables extended interactions, personalization, temporal awareness, and coherence across disjoint sessions or long tasks. In practice, many deployed systems go beyond basic retrieval-augmented generation by using structured memory representations and explicit memory operations (write, update, compress, forget), improving retrieval efficiency and reducing redundancy.

*7.1.1 Dialogue Assistants.* In general-purpose chatbots and voice assistants, long-term memory is most often realized as a persistent user profile and interaction-history store: stable preferences (e.g., formatting, tone), constraints (e.g., dietary restrictions), and durable facts (e.g., recurring projects) are written over time and retrieved selectively to personalize responses across sessions. This creates a practical separation between a compact working context and a larger long-term store, where only the most relevant memories are surfaced under strict context limits. Representative systems span both end-user products and developer-facing frameworks. Productized assistants increasingly expose persistent memory as a controllable feature, such as **ChatGPT Memory**[2] and the **Claude memory tool**[3], where retention and deletion policies help mitigate stale or sensitive facts. On the open-source side, **MemGPT**[4] [114] implements an explicit memory manager that mediates what is stored and what is retrieved, while common orchestration stacks (e.g., **LangChain Memory**[5] and **Rasa**[6] tracker stores) provide persistent dialogue state and configurable backends. **BaiJia MemoryOS**[7] is designed to provide a memory operating system for personalized AI agents, enabling more coherent, personalized, and context-aware interactions.

*7.1.2 Embodied Robotics.* Embodied agents (e.g., household robots) must operate over long horizons while continuously perceiving a changing world. Long-term memory therefore acts as a persistent world model: a record of observations (vision/sensors), inferred object states, and action histories that supports planning without replaying the full interaction trace each time. Unlike purely textual assistants, robotics memory is intrinsically multimodal and spatial: queries must resolve to concrete entities (objects/places) and time (last seen, last manipulated), which makes indexing and update latency central to system utility. Representative systems illustrate complementary realizations of this memory. In open-world embodied settings such as **Voyager**[8], agents accumulate procedural memory (skill libraries) alongside episodic traces to improve long-horizon competence. In non-LLM but practically deployed robotics stacks, **Home-Robot**[9] provides a persistent semantic mapping pipeline that maintains object-centric spatial memory for navigation and manipulation.

*7.1.3 Personalized Recommendation Systems.* Recommendation systems rely on long-term memory as a persistent interaction history and preference representation: clicks, watches, purchases, and feedback are stored with timestamps to model evolving interests and to support recency-aware ranking. In memory-enhanced pipelines, the key operational concern is not only storing more history, but controlling how history is summarized, decayed, and surfaced so the system can both adapt to preference drift and provide intelligible explanations. Beyond research prototypes, several widely used platforms expose these memory capabilities as services. **Amazon Personalize**[10] operationalizes event-history ingestion and user/item representations for real-time personalization. **Google Gemini**[11] generates tailored, context-aware recommendations by leveraging users' conversation history and its deep integration with Google's ecosystem. In all personalized recommendation applications, privacy protection based on user memory is critical, such as browsing histories, purchase records, preference profiles, and interaction patterns.

[2] https://help.openai.com/en/articles/8590148-memory-faq
[3] https://platform.claude.com/docs/en/agents-and-tools/tool-use/memory-tool
[4] https://github.com/cpacker/MemGPT
[5] https://python.langchain.com/docs/modules/memory/
[6] https://github.com/RasaHQ/rasa
[7] https://baijia.online/memoryos/
[8] https://github.com/MineDojo/Voyager
[9] https://github.com/facebookresearch/home-robot
[10] https://aws.amazon.com/personalize/
[11] https://deepmind.google/models/gemini/

*7.1.4 Medical Diagnosis.* Clinical AI assistants require long-term memory as a faithful longitudinal patient record store: diagnoses, medications, allergies, test results, and clinician notes often span years and must be recalled reliably to avoid unsafe omissions. Compared with consumer applications, medical memory places unusually strict demands on completeness, auditability, and provenance, because incorrect recall can directly impact patient safety and clinical accountability. Representative systems cover both open-source predictive models and widely used patient-facing tools. **Doctor AI**[12] [25] illustrates longitudinal modeling over multi-visit EHR sequences, where historical trajectories function as memory for risk prediction. For consumer triage, **Ada Health**[13] and **Buoy Health**[14] maintain user-reported symptom histories and profile attributes across interactions to improve continuity and reduce repetitive questioning. While many LLM-based clinical assistants remain research-facing, practical deployments consistently converge on the same memory principles: explicit clinician queries, provenance-linked retrieval, and conservative update policies that keep prior facts available for review rather than silently overwriting them.

*7.1.5 Content Generation and Creative Writing.* In long-form creative generation (storytelling, role-playing, interactive fiction), memory typically appears as a persistent story bible: character profiles, world rules, relationship graphs, and event timelines that constrain future generations [61]. The most effective systems mix multiple memory granularities by using high-level summaries for global coherence and fine-grained facts for continuity, so that generation can remain creative while avoiding contradictions. A number of user-facing writing systems expose these memory mechanisms directly. **AI Dungeon**[15] and **NovelAI**[16] provide dedicated memory and lore features that store canonical facts and retrieve them during generation, often allowing authors to curate what the model must preserve. In the open-source ecosystem, **SillyTavern**[17] operationalizes long-running character chats with configurable lorebooks and optional vector retrieval, making memory both editable and inspectable by users. Character-centric platforms such as **Character.AI**[18] and **BaiJia Agents**[19] historical agents platform similarly rely on persistent personas and long-running conversation traces to sustain role-playing identity over time.

### 7.2 Multi Agent Memory Applications

In multi-agent AI applications, memory supports shared task state, coordination, and experience reuse across multiple roles. Compared with single-agent settings, memory must additionally address concurrency, provenance, and access control, because multiple agents may read or write the same artifacts. Interactions are often explicit at the agent level (through shared workspaces or memory APIs), while user visibility depends on the application domain, safety posture, and organizational governance.

*7.2.1 Collaborative Workflows.* In team workflows (e.g., software engineering, analysis, or content production), shared memory is most naturally instantiated as a workspace: plans, intermediate artifacts, tool outputs, and decisions are stored in a common substrate so specialized agents can coordinate without tight synchronization. This workspace typically behaves like a versioned project state, where agents both consume prior context and contribute new artifacts that downstream agents depend on. Representative frameworks operationalize this design with explicit read and write

[12] https://github.com/mp2893/doctorai
[13] https://ada.com/
[14] https://www.buoyhealth.com/
[15] https://play.aidungeon.com/
[16] https://novelai.net/
[17] https://github.com/SillyTavern/SillyTavern
[18] https://character.ai/
[19] https://baijia.online/home

primitives and inspectable traces. **MetaGPT**[20] [54] and **AutoGen**[21] [166] demonstrate role-based orchestration where shared artifacts (specs, code, reports) serve as the durable coordination medium. **CrewAI**[22] provides a lightweight production-oriented abstraction for agent teams with shared context, while **LangGraph**[23] emphasizes stateful agent graphs where memory is treated as an explicit, evolvable state machine. In developer tooling, systems such as **OpenDevin**[24] illustrate how an agent team can share a filesystem-like workspace plus logs for provenance. Across these systems, concurrency control (locking, merging, or turn-taking), conflict resolution, and provenance tracking (who produced what, when, and from which evidence) are essential to prevent stale or contradictory shared state from contaminating downstream decisions.

*7.2.2 Open-World Simulation.* In open-world simulations, memory enables coherent long-term social behavior by recording both shared events (world logs) and private experiences (agent-specific episodic traces). Agents retrieve salient past interactions to choose actions that reflect continuity (e.g., remembering promises, rivalries, shared activities), while the simulation remains consistent by anchoring multiple agents to the same event substrate. **Generative Agents**[25] [116] exemplify this approach by maintaining per-agent memories and retrieving them via salience and recency to drive daily routines and social interactions. **AI Town**[26] provides a practical, runnable environment that extends this paradigm into an interactive sandbox, where world events and agent memories co-evolve as the simulation progresses. Operationally, these systems rely on salience-based retention to avoid storing everything, explicit separation of private versus shared memory to prevent unintended information leakage, and temporal consistency mechanisms so that the same shared event is not reinterpreted incompatibly across agents. Memory interaction is largely implicit to observers, but explicit in the simulation engine through continual logging and retrieval.

*7.2.3 Scientific Discovery.* Scientific multi-agent systems use shared memory as a knowledge base with provenance: extracted facts, hypotheses, intermediate results, and tool outputs must be stored in structured forms so that agents can build on one another's work while preserving verifiability. Unlike generic collaboration, scientific workflows often require schema constraints (entities, measurements, units, citations) and human inspectable records so domain experts can audit and correct the evolving memory. Representative systems implement this by coupling tool-driven extraction with durable repositories. **SciAgents**[27] [44] and **MRAgent**[28] [172] emphasize structured records and provenance so that hypotheses and evidence remain traceable across iterations. In practice, many research agent stacks also incorporate shared literature memory. For example, **PaperQA**[29] provides a persistent repository grounded in papers that can be reused across questions and agents. Across these systems, memory must be both precise (schema aligned, citation linked) and robust to revision, because scientific progress often requires revisiting and updating prior beliefs as new evidence arrives.

*7.2.4 Financial Decision Support.* Financial multi-agent systems use shared memory to maintain evolving belief states and evidence logs under nonstationary markets. Here, memory is simultaneously time sensitive and audit sensitive:

[20] https://github.com/FoundationAgents/MetaGPT
[21] https://github.com/microsoft/autogen
[22] https://github.com/crewAIInc/crewAI
[23] https://github.com/langchain-ai/langgraph
[24] https://github.com/OpenDevin/OpenDevin
[25] https://github.com/joonspk-research/generative_agents
[26] https://github.com/a16z-infra/ai-town
[27] https://github.com/lamm-mit/SciAgentsDiscovery
[28] https://github.com/xuwei1997/MRAgent
[29] https://github.com/whitead/paper-qa

agents must prioritize fresh signals while retaining enough historical rationale to justify decisions, support risk controls, and enable post hoc review. In open source ecosystems, **FinGPT**[30] and **OpenBB**[31] provide practical building blocks for constructing finance agents that maintain persistent stores of retrieved news, indicators, analysis notes, and decision rationales. In this domain, freshness control, low-latency updates, and strong traceability (linking each recommendation to retrieved evidence and the state of shared beliefs at the time) are necessary to balance responsiveness with accountability.

*7.2.5 Experience Sharing Platform.* Experience accumulation and cross-agent reuse constitute the core driving force for scaling the capabilities of AI systems from individual intelligence to collective intelligence. Establishing a robust experience sharing platform is therefore a pivotal step towards breaking the knowledge silos among isolated agents, facilitating collaborative learning, and enabling the systematic evolution of general-purpose AI systems. Amid the rapid advancements in large model memory management, most existing frameworks focus solely on individual-level memory optimization, leaving a critical research gap in the standardized storage, cross-user retrieval, and collective reuse of group-level experiential memory. Addressing this gap, **MemoryOS++**[1] emerges as the first dedicated platform for group-level experiential memory sharing, pioneering a unified paradigm to scale experience reuse across heterogeneous users and AI agents. Built on **MemoryOS**[73], an operating system memory framework for individual memory management with a hierarchical storage architecture and four core modules (Storage, Updating, Retrieval, Generation) that handle personal interaction histories, user profiles, and domain knowledge, MemoryOS++ further develops a full-fledged shared experience platform for collaborative problem-solving. Leveraging MemoryOS's robust personal memory foundation, it introduces a unified memory sharing mechanism to standardize storage formats, indexing operations, and retrieval protocols for experiential memory across diverse users and agents. By establishing cross-user shared experience pools and incorporating dedicated mechanisms for experience value measurement and experience fusion, MemoryOS++ empowers AI agents to retrieve multi-perspective human experiences and replicate proven solution paths in analogous task contexts. This innovative design fundamentally elevates memory from a fragmented individual capability to a centralized, group-level resource, laying a critical foundation for large-scale collective learning and experience reuse in AI systems. Most notably, it culminates in the realization of swarm intelligence, marking a transformative leap from isolated individual memory utilization to synergistic group-level cognitive collaboration.

## 8 Challenges and Future of AI Memory

By bridging limited context windows with infinite information streams, memory enables long-term consistency, complex reasoning, self-evolution, and lifelong learning. We analyze critical bottlenecks in current memory-augmented agents and chart the trajectory for next-generation memory architectures from two dimensions:

- **Challenges in Current AI Memory (§8.1)**: We dissect the multidimensional obstacles constraining the field, systematically categorizing them into architectural conflicts, theoretical gaps, and operational security risks.
- **Future Trends of AI Memory (§8.2)**: We envision the evolution from passive retrieval to active adaptive systems, highlighting brain-inspired mechanisms, the transition from memory to experience, and collective memory frameworks.

[30] https://github.com/AI4Finance-Foundation/FinGPT
[31] https://github.com/OpenBB-finance/OpenBB

### 8.1 Challenges in Current AI Memory

Despite the significant progress in autonomous agents, the development of robust memory systems encounters multidimensional obstacles that hinder broad application. These challenges span from intrinsic model limitations to external deployment risks. In this section, we classify these impediments into three main categories: architectural conflicts regarding system design, theoretical gaps in methodology and evaluation, and the security complexities inherent in practical operations.

*8.1.1 Architectural Conflicts and System Limitations.* A fundamental conflict arises between the inherent characteristics of LLMs and the functional requirements of effective memory systems. First, the finite context window of LLMs struggles to accommodate the massive accumulation of long-term experiences. Although context windows have expanded, they cannot substitute for persistent storage, and externally stored memory systems often face bottlenecks in retrieval efficiency and semantic relevance matching [114]. Furthermore, the parametric memory update process presents a significant dilemma. Solidifying new experiences into model parameters via fine-tuning is computationally expensive and highly susceptible to catastrophic forgetting, where the model loses previously acquiring knowledge while adapting to dynamic information [110]. Finally, the integration of multimodal memory introduces substantial complexity. Agents are required to process text, images, speech, and sensor data simultaneously, yet unifying these heterogeneous information sources into a coherent and storable memory representation remains a formidable technical challenge [69].

*8.1.2 Theoretical and Methodological Challenges.* From a theoretical research standpoint, several conceptual and methodological obstacles remain unresolved. One primary issue is the incomplete understanding of memory itself. Many existing studies focus predominantly on the temporal dimension of memory, such as recency and longevity, while paying insufficient attention to the object dimension and the storage form. This imbalance leads to a fragmented conceptualization that fails to capture the full complexity of cognitive processes. Besides, memory evaluation presents another significant hurdle. Existing benchmarks emphasize basic metrics like memory capacity and retrieval accuracy but lack systematic standards for higher-level properties. There is a notable absence of metrics for assessing memory generalization across diverse tasks, robustness against noise, multimodal consistency, and the quality of human-centric interaction. Additionally, the field lacks a mature theoretical framework for multi-agent memory sharing. Critical mechanisms for partitioning and synchronizing information among multiple agents remain underexplored, particularly regarding how to maintain consistency, assign credit, and ensure scalability in decentralized environments [54, 116].

*8.1.3 Security Risks and Operational Complexities.* In practical deployment, memory implementation faces deep-seated challenges regarding security, privacy, and operational complexity. For personal agents, the storage of user data necessitates a rigorous balance between personalization and privacy preservation. The risk extends beyond data leakage, as powerful models might inadvertently infer sensitive attributes from seemingly innocuous memory records, making robust access control a prerequisite for commercialization [89]. In collaborative scenarios involving memory sharing, the difficulties are compounded by the need for dynamic governance. Existing solutions often adopt static permission designs that are ill-suited for complex environments. Without adaptive mechanisms for permission allocation, conflict resolution, and real-time updates, multi-agent systems are prone to efficiency bottlenecks and data inconsistency risks, which directly undermine the reliability of autonomous collaboration [29, 54].

### 8.2 Future Trends of AI Memory

To advance lifelong autonomy, memory architectures must evolve from passive retrieval to active adaptive systems. This section highlights three pivotal directions: brain-inspired mechanisms addressing the stability-plasticity dilemma and multimodal integration; the transition from memory to experience to drive self-evolution through structured priors; and self-evolving collective memory for secure, robust multi-agent coordination.

*8.2.1 Brain-Inspired Memory Modeling.* Building robust, lifelong agents requires architectural insights from the biological brain to overcome the stability-plasticity dilemma and limitations in multimodal integration [78]. Biological networks address these challenges by treating memory as a dynamic, multi-timescale process.

- **Biomimetic Long-Term Memory Consolidation.** Drawing from Complementary Learning Systems (CLS) theory [80, 106], future architectures should model the separation of rapid acquisition from gradual integration. This dual-structure approach addresses the stability-plasticity dilemma by distinguishing a volatile buffer for recent interactions from a stable repository. Instead of expensive real-time updates, agents can employ learnable, scheduled offline phases to synthesize raw logs, effectively decoupling heavy updates from online inference as demonstrated by architectures like LightMem [38]. Such mechanisms ensure the long-term store remains compact while allowing agents to learn quickly from specific experiences without disturbing established regularities.
- **Unified Multimodal Memory.** Biological memory relies on ensembles that integrate sensory signals into stable joint representations, implying that perception, memory, and action should share substrates [41]. Future agents could adopt this holistic approach by constructing 'multimodal events' where sensory, linguistic, and environmental states are semantically linked. To achieve this, architectures are evolving towards entity-centric graphs and hierarchical knowledge bases that bind perception directly to execution [91, 99]. By mapping heterogeneous inputs into shared formats, these systems enable cross-modal retrieval where visual cues seamlessly trigger executable action plans, forming a unified fabric of experience.

*8.2.2 From AI Memory to AI Experience.* Advancing from memory to experience requires not only shifting from the passive retrieval of traditional RAG to active, goal-oriented utilization but also expanding from individual-level memory to group-level experiential intelligence, a transition enabled by platforms like MemoryOS++[1], which is a pivotal platform dedicated to swarm experiential memory sharing. Beyond that, this evolution entails organizing unstructured logs into structured priors that directly guide reasoning and planning, ensuring that past interactions continuously optimize future behavior and drive self-evolution.

- **From Static Logs to Goal-Aware Memory.** A fundamental step toward deep synergy involves moving from unstructured logs to explicitly structured episodes with inherent temporal and causal dependencies [116, 117]. Such structure enables memory management to evolve from passive storage into an active policy, where agents primarily frame maintenance as a sequential decision problem to filter and synthesize information [155]. Consequently, retrieval strategies must shift from fixed similarity lookups to learned, goal-aware decisions. By transforming retrieval into a dynamic evaluation process, agents can prioritize memories that best reduce uncertainty rather than relying solely on surface-level matching [173], providing the high-quality structured priors necessary for complex cognition.
- **Experience-Driven Reasoning and Planning.** Leveraging these structured priors, agents can transcend simple context augmentation to actively shape reasoning and planning [81]. In reasoning tasks, stored episodes

serve as experiential precedents to guide internal search. This mechanism suggests decompositions or pruning branches to form a coupled process where retrieval informs the chain of thought [131, 182]. For planning, episodic traces can be distilled into executable schemas or workflow templates stored in semantic memory. Agents can thus retrieve and adapt these experiential templates to reduce search costs instead of synthesizing plans from scratch [158]. This utilization completes a closed feedback loop where executed plans generate new trajectories for consolidation, transforming accumulated logs into a curated repertoire of strategies that continuously enhance future performance [190].

*8.2.3 Self-Evolving Collective Memory.* As AI shifts to multi-agent systems, shared memory must transcend simplistic global blackboards to address vulnerability and noise accumulation. This section examines mechanisms for jointly maintaining a safe and efficient collective memory through dynamic access control and autonomous optimization.

- **Dynamic Permissions and Access Control.** Shared memory requires a shift from static access control to dynamic, context-aware protocols that accommodate emergent collaborative needs [125]. Access rights should depend on evolving states and trust, necessitating the integration of Theory of Mind to model requester intent alongside credentials [86]. Furthermore, control mechanisms must incorporate privacy-preserving abstractions such as differential privacy. This allows agents to share statistical insights for global optimization without exposing raw episodes [191], effectively balancing the trade-off between open collaboration and data security.
- **Self-Organization and Consensus.** To prevent shared memory from degenerating into noisy redundancy, collective memory must function as a self-organizing knowledge base. This requires automatic de-duplication and clustering mechanisms to consolidate semantically similar entries into canonical nodes, thereby reducing entropy [138, 183]. Beyond structure, active consensus protocols are essential for resolving conflicts and managing maintenance. By employing voting mechanisms like adapted PBFT, agents can jointly execute collaborative forgetting to prune low-utility data [6, 17, 184], ensuring the collective store remains both accurate and compact through an autonomous optimization layer.

## 9 Conclusion

This survey comprehensively reviews AI memory mechanisms in LLM-based agents, covering theoretical foundations in cognitive science to practical implementations in complex application scenarios. By bridging biological memory models and computational architectures, we clarify that memory is not merely static storage but a dynamic cognitive substrate critical for continuous learning and adaptation. To systematically organize the burgeoning literature, we propose the 4W Memory Taxonomy, which characterizes memory design via four core dimensions (When-What-How-Which). This taxonomy provides a structured framework to decompose complex agent architectures and standardize field discourse. Based on this taxonomy, we further analyzed the operation of these mechanisms in single- and multi-agent systems. We also systematically synthesize existing evaluation benchmarks for AI memory, which lay a fundamental basis for standardized assessment in this field.

Looking forward, the evolution of agent memory marks a pivotal milestone on the path toward Artificial General Intelligence. We envision AI memory evolving beyond rudimentary information retrieval to emerge as the driving engine behind autonomous cognitive evolution. Future research will likely focus on two core thrusts: brain-inspired memory consolidation mechanisms and privacy-preserving collective memory protocols. Ultimately, robust AI memory systems will enable agents to evolve from static tools into adaptive, lifelong collaborative companions, ushering in a new era of symbiotic co-evolution between human cognition and artificial cognition.

## References


[1] Kwangseob Ahn. 2025. HEMA: A Hippocampus-Inspired Extended Memory Architecture for Long-Context AI Conversations. *arXiv preprint arXiv:2504.16754* (2025).

[2] Qingyao Ai, Yichen Tang, Changyue Wang, Jianming Long, Weihang Su, and Yiqun Liu. 2025. MemoryBench: A Benchmark for Memory and Continual Learning in LLM Systems. *arXiv preprint arXiv:2510.17281* (2025).

[3] Petr Anokhin, Nikita Semenov, Artyom Sorokin, Dmitry Evseev, Andrey Kravchenko, Mikhail Burtsev, and Evgeny Burnaev. 2024. Arigraph: Learning knowledge graph world models with episodic memory for llm agents. *arXiv preprint arXiv:2407.04363* (2024).

[4] Anonymous. 2025. MemoryField: Exploiting Gravitational Field for Long-term Memory Management. In *Submitted to The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=KpaKtbGgUa under review.

[5] Richard C Atkinson and Richard M Shiffrin. 1968. Human memory: A proposed system and its control processes. In *Psychology of learning and motivation*. Vol. 2. Elsevier, 89–195.

[6] Duong Bach. 2025. PBFT-Backed Semantic Voting for Multi-Agent Memory Pruning. *arXiv preprint arXiv:2506.17338* (2025).

[7] Alan Baddeley. 2000. The episodic buffer: a new component of working memory? *Trends in cognitive sciences* 4, 11 (2000), 417–423.

[8] Alan Baddeley. 2012. Working memory: Theories, models, and controversies. *Annual review of psychology* 63, 1 (2012), 1–29.

[9] Alan D Baddeley and Graham Hitch. 1974. Working memory. In *The psychology of learning and motivation*. Vol. 8. Elsevier, 47–89.

[10] Ting Bai, Le Huang, Yue Yu, Cheng Yang, Cheng Hou, Zhe Zhao, and Chuan Shi. 2025. Efficient multi-task prompt tuning for recommendation. *ACM Transactions on Information Systems* 43, 4 (2025), 1–21.

[11] Ting Bai, Jiazheng Kang, and Jiayang Fan. 2024. BaiJia: A Large-Scale Role-Playing Agent Corpus of Chinese Historical Characters. *arXiv preprint arXiv:2412.20024* (2024).

[12] Ting Bai, Yue Yu, Le Huang, Zenan Xu, and Chuan Shi. 2024. GMoE: Empowering LLMs Fine-Tuning via MoE Graph Collaboration. *arXiv preprint arXiv:2412.16216* (2024).

[13] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. 2024. Longbench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*. 3119–3137.

[14] Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, et al. 2025. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 3639–3664.

[15] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[16] Jiaqi Cao, Jiarui Wang, Rubin Wei, Qipeng Guo, Kai Chen, Bowen Zhou, and Zhouhan Lin. 2025. Memory decoder: A pretrained, plug-and-play memory for large language models. *arXiv preprint arXiv:2508.09874* (2025).

[17] Miguel Castro, Barbara Liskov, et al. 1999. Practical byzantine fault tolerance. In *OsDI*, Vol. 99. 173–186.

[18] Mert Cemri, Melissa Z Pan, Shuyi Yang, Lakshya A Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, et al. 2025. Why do multi-agent llm systems fail? *arXiv preprint arXiv:2503.13657* (2025).

[19] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology* 15, 3 (2024), 1–45.

[20] Ding Chen, Simin Niu, Kehang Li, Peng Liu, Xiangping Zheng, Bo Tang, Xinchi Li, Feiyu Xiong, and Zhiyu Li. 2025. HaluMem: Evaluating Hallucinations in Memory Systems of Agents. *arXiv preprint arXiv:2511.03506* (2025).

[21] Guo Chen, Yifei Huang, Yin-Dong Zheng, Yicheng Liu, Jiahao Wang, and Tong Lu. 2025. Egocentric object-interaction anticipation with retentive and predictive learning. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*. 783–791.

[22] Weijie Chen, Ting Bai, Jinbo Su, Jian Luan, Wei Liu, and Chuan Shi. 2024. Kg-retriever: Efficient knowledge indexing for retrieval-augmented large language models. *arXiv preprint arXiv:2412.05547* (2024).

[23] Yinpeng Chen, DeLesley Hutchins, Aren Jansen, Andrey Zhmoginov, David Racz, and Jesper Andersen. 2024. Melodi: Exploring memory compression for long contexts. *arXiv preprint arXiv:2410.03156* (2024).

[24] 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* (2025).

[25] Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F Stewart, and Jimeng Sun. 2016. Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine learning for healthcare conference*. PMLR, 301–318.

[26] Nelson Cowan. 2001. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and brain sciences* 24, 1 (2001), 87–114.

[27] Jisheng Dang, Huicheng Zheng, Xudong Wu, Jingmei Jiao, Bimei Wang, Jun Yang, Bin Hu, Jianhuang Lai, and Tat Seng Chua. 2025. External Memory Matters: Generalizable Object-Action Memory for Retrieval-Augmented Long-Term Video Understanding. In *34th Internationa Joint Conference on Artificial Intelligence, IJCAI 2025*. International Joint Conferences on Artificial Intelligence, 864–872.

[28] Christopher J Darwin, Michael T Turvey, and Robert G Crowder. 1972. An auditory analogue of the Sperling partial report procedure: Evidence for brief auditory storage. *Cognitive Psychology* 3, 2 (1972), 255–267.

[29] Chad DeChant. 2025. Episodic memory in ai agents poses risks that should be studied and mitigated. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 321–332.

[30] Yang Deng, Xuan Zhang, Wenxuan Zhang, Yifei Yuan, See Kiong Ng, and Tat-Seng Chua. 2024. On the multi-turn instruction following for conversational web agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 8795–8812.

[31] Matthijs Dopacker2023memgptuze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). arXiv:2401.08281 [cs.LG]

[32] Yiming Du, Wenyu Huang, Danna Zheng, Zhaowei Wang, Sebastien Montella, Mirella Lapata, Kam-Fai Wong, and Jeff Z Pan. 2025. Rethinking memory in ai: Taxonomy, operations, topics, and future directions. *arXiv preprint arXiv:2505.00675* (2025).

[33] Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. 2024. PerLTQA: A personal long-term memory dataset for memory classification, retrieval, and fusion in question answering. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*. 152–164.

[34] Zhuoyun Du, Runze Wang, Huiyu Bai, Zouying Cao, Xiaoyong Zhu, Bo Zheng, Wei Chen, and Haochao Ying. 2025. Enabling Agents to Communicate Entirely in Latent Space. *arXiv preprint arXiv:2511.09149* (2025).

[35] Harsh Dubey and Chulwoo Pack. 2025. Leveraging Textual Memory and Key Frame Reasoning for Full Video Understanding Using Off-the-Shelf LLMs and VLMs (Student Abstract). In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 29351–29352.

[36] Yue Fan, Xiaojian Ma, Rongpeng Su, Jun Guo, Rujie Wu, Xi Chen, and Qing Li. 2025. Embodied videoagent: Persistent memory from egocentric videos and embodied sensors enables dynamic scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6342–6352.

[37] Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. 2024. Videoagent: A memory-augmented multimodal agent for video understanding. In *European Conference on Computer Vision*. Springer, 75–92.

[38] Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, et al. 2025. Lightmem: Lightweight and efficient memory-augmented generation. *arXiv preprint arXiv:2510.18866* (2025).

[39] Zafeirios Fountas, Martin Benfeghoul, Adnan Oomerjee, Fenia Christopoulou, Gerasimos Lampouras, Haitham Bou Ammar, and Jun Wang. 2025. Human-inspired episodic memory for infinite context LLMs. In *The Thirteenth International Conference on Learning Representations*.

[40] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. 2025. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 24108–24118.

[41] Joaquín M Fuster. 2009. Cortex and memory: emergence of a new paradigm. *Journal of cognitive neuroscience* 21, 11 (2009), 2047–2072.

[42] Hang Gao and Yongfeng Zhang. 2024. Memory sharing for large language model based agents. *arXiv preprint arXiv:2404.09982* (2024).

[43] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Commun. ACM* 64, 12 (2021), 86–92.

[44] Alireza Ghafarollahi and Markus J Buehler. 2025. SciAgents: automating scientific discovery through bioinspired multi-agent intelligent graph reasoning. *Advanced Materials* 37, 22 (2025), 2413523.

[45] Gabrielle Girardeau, Karim Benchenane, Sidney I Wiener, György Buzsáki, and Michaël B Zugaro. 2009. Selective suppression of hippocampal ripples impairs spatial memory. *Nature neuroscience* 12, 10 (2009), 1222–1223.

[46] Significant Gravitas. 2023. AutoGPT: An Autonomous GPT-4 Experiment. https://github.com/Significant-Gravitas/AutoGPT.

[47] Tianle Gu, Kexin Huang, Ruilin Luo, Yuanqi Yao, Xiuying Chen, Yujiu Yang, Yan Teng, and Yingchun Wang. 2025. From Evasion to Concealment: Stealthy Knowledge Unlearning for LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*. 10261–10279.

[48] Jing Guo, Nan Li, Jianchuan Qi, Hang Yang, Ruiqiao Li, Yuzhen Feng, Si Zhang, and Ming Xu. 2023. Empowering working memory for large language model agents. *arXiv preprint arXiv:2312.17259* (2023).

[49] Nilesh Gupta, Wei-Cheng Chang, Ngot Bui, Cho-Jui Hsieh, and Inderjit S Dhillon. 2025. LLM-guided Hierarchical Retrieval. *arXiv preprint arXiv:2510.13217* (2025).

[50] Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From rag to memory: Non-parametric continual learning for large language models. *arXiv preprint arXiv:2502.14802* (2025).

[51] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. 2024. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13504–13514.

[52] Zifan He, Yingqi Cao, Zongyue Qin, Neha Prakriya, Yizhou Sun, and Jason Cong. 2025. Hmt: Hierarchical memory transformer for efficient long context language processing. 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)*. 8068–8089.

[53] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. 2021. Knowledge graphs. *ACM Computing Surveys (Csur)* 54, 4 (2021), 1–37.

[54] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, et al. 2023. MetaGPT: Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations*.

[55] Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: What's the Real Context Size of Your Long-Context Language Models? *arXiv preprint arXiv:2404.06654* (2024).

[56] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv preprint arXiv:2106.09685* (2021).

[57] Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. 2025. Hiagent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 32779–32798.

[58] Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. 2024. Unlearning or obfuscating? jogging the memory of unlearned llms via benign relearning. *arXiv preprint arXiv:2406.13356* (2024).

[59] Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 2025. 3DLLM-Mem: Long-Term Spatial-Temporal Memory for Embodied 3D Large Language Model. *arXiv preprint arXiv:2505.22657* (2025).

[60] Yuanzhe Hu, Yu Wang, and Julian McAuley. 2025. Evaluating memory in llm agents via incremental multi-turn interactions. *arXiv preprint arXiv:2507.05257* (2025).

[61] Le Huang, Hengzhi Lan, Zijun Sun, Chuan Shi, and Ting Bai. 2024. Emotional RAG: Enhancing role-playing agents through emotional retrieval. In *2024 IEEE International Conference on Knowledge Graph (ICKG)*. IEEE, 120–127.

[62] Shijue Huang, Wanjun Zhong, Deng Cai, Fanqi Wan, Chengyi Wang, Mingxuan Wang, Mu Qiao, and Ruifeng Xu. 2025. Empowering Self-Learning of LLMs: Inner Knowledge Explicitation as a Catalyst. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 24150–24158.

[63] Zhengjun Huang, Zhoujin Tian, Qintian Guo, Fangyuan Zhang, Yingli Zhou, Di Jiang, and Xiaofang Zhou. 2025. LiCoMemory: Lightweight and Cognitive Agentic Memory for Efficient Long-Term Reasoning. *arXiv preprint arXiv:2511.01448* (2025).

[64] Alexis Huet, Zied Ben Houidi, and Dario Rossi. 2025. Episodic memories generation and evaluation benchmark for large language models. *arXiv preprint arXiv:2501.13121* (2025).

[65] Jitesh Jain, Shubham Maheshwari, Ning Yu, Wen-mei Hwu, and Humphrey Shi. 2025. AUGUSTUS: An LLM-Driven Multimodal Agent System with Contextualized User Memory. *arXiv preprint arXiv:2510.15261* (2025).

[66] Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J Taylor, and Dan Roth. 2025. Know me, respond to me: Benchmarking llms for dynamic user profiling and personalized responses at scale. *arXiv preprint arXiv:2504.14225* (2025).

[67] Hongda Jiang, Xinyuan Zhang, Siddhant Garg, Rishab Arora, Shiun-Zu Kuo, Jiayang Xu, Aaron Colak, and Xin Luna Dong. 2025. Memory-QA: Answering Recall Questions Based on Multimodal Memories. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 24255–24277.

[68] Xun Jiang, Feng Li, Han Zhao, Jiahao Qiu, Jiaying Wang, Jun Shao, Shihao Xu, Shu Zhang, Weiling Chen, Xavier Tang, et al. 2024. Long term memory: The foundation of ai self-evolution. *arXiv preprint arXiv:2410.15665* (2024).

[69] Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. Hipporag: Neurobiologically inspired long-term memory for large language models. *Advances in Neural Information Processing Systems* 37 (2024), 59532–59569.

[70] Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Rwku: Benchmarking real-world knowledge unlearning for large language models. *Advances in Neural Information Processing Systems* 37 (2024), 98213–98263.

[71] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.

[72] Jiazheng Kang, Le Huang, Cheng Hou, Zhe Zhao, Zhenxiang Yan, and Ting Bai. 2025. Self-Evolving LLMs via Continual Instruction Tuning. *arXiv preprint arXiv:2509.18133* (2025).

[73] Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. 2025. Memory OS of AI Agent. *arXiv preprint arXiv:2506.06326* (2025).

[74] Jiho Kim, Woosog Chay, Hyeonji Hwang, Daeun Kyung, Hyunseung Chung, Eunbyeol Cho, Yohan Jo, and Edward Choi. 2024. Dialsim: A real-time simulator for evaluating long-term dialogue understanding of conversational agents. *arXiv e-prints* (2024), arXiv–2406.

[75] 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* (2025).

[76] Jens G Klinzing, Niels Niethard, and Jan Born. 2019. Mechanisms of systems memory consolidation during sleep. *Nature neuroscience* 22, 10 (2019), 1598–1610.

[77] Ishant Kohar and Aswanth Krishnan. 2025. A Benchmark for Procedural Memory Retrieval in Language Agents. *arXiv preprint arXiv:2511.21730* (2025).

[78] Dhireesha Kudithipudi, Mario Aguilar-Simon, Jonathan Babb, Maxim Bazhenov, Douglas Blackiston, Josh Bongard, Andrew P Brna, Suraj Chakravarthi Raja, Nick Cheney, Jeff Clune, et al. 2022. Biological underpinnings for lifelong learning machines. *Nature Machine Intelligence* 4, 3 (2022), 196–210.

[79] Aman Kumar, Ekant Muljibhai Amin, Xian Yeow Lee, Lasitha Vidyaratne, Ahmed K Farahat, Dipanjan D Ghosh, Yuta Koreeda, and Chetan Gupta. 2025. Building Domain-Specific Small Language Models via Guided Data Generation. *arXiv preprint arXiv:2511.21748* (2025).

[80] Dharshan Kumaran, Demis Hassabis, and James L McClelland. 2016. What learning systems do intelligent agents need? Complementary learning systems theory updated. *Trends in cognitive sciences* 20, 7 (2016), 512–534.

[81] Hengzhi Lan, Yue Yu, Li Qian, Li Peng, Jie Wu, Wei Liu, Jian Luan, and Ting Bai. 2025. LightSearcher: Efficient DeepSearch via Experiential Memory. *arXiv preprint arXiv:2512.06653* (2025).

[82] Hung Le, Kien Do, Dung Nguyen, Sunil Gupta, and Svetha Venkatesh. 2024. Stable Hadamard Memory: Revitalizing Memory-Augmented Agents for Reinforcement Learning. *arXiv preprint arXiv:2410.10132* (2024).

[83] Mingcong Lei, Yiming Zhao, Ge Wang, Zhixin Mai, Shuguang Cui, Yatong Han, and Jinke Ren. 2025. STMA: A spatio-temporal memory agent for long-horizon embodied task planning. *arXiv preprint arXiv:2502.10177* (2025).

[84] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems* 33 (2020), 9459–9474.

[85] Dongfang Li, Zetian Sun, Xinshuo Hu, Baotian Hu, and Min Zhang. 2025. Cmt: A memory compression method for continual knowledge learning of large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 24413–24421.

[86] Huao Li, Yu Chong, Simon Stepputtis, Joseph P Campbell, Dana Hughes, Charles Lewis, and Katia Sycara. 2023. Theory of mind for multi-agent collaboration via large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 180–192.

[87] Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. 2025. 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)*. 5259–5276.

[88] Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. 2024. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218* (2024).

[89] Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, et al. 2024. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459* (2024).

[90] Zhiyu Li, Shichao Song, Chenyang Xi, Hanyu Wang, Chen Tang, Simin Niu, Ding Chen, Jiawei Yang, Chunyu Li, Qingchen Yu, et al. 2025. Memos: A memory os for ai system. *arXiv preprint arXiv:2507.03724* (2025).

[91] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. 2024. Optimus-1: Hybrid multimodal memory empowered agents excel in long-horizon tasks. *Advances in neural information processing systems* 37 (2024), 49881–49913.

[92] Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, et al. 2023. Mm-vid: Advancing video understanding with gpt-4v (ision). *arXiv preprint arXiv:2310.19773* (2023).

[93] Yueqian Lin, Qinsi Wang, Hancheng Ye, Yuzhe Fu, Hai Li, Yiran Chen, et al. 2025. Hippomm: Hippocampal-inspired multimodal memory for long audiovisual event understanding. *arXiv preprint arXiv:2504.10739* (2025).

[94] Akide Liu, Jing Liu, Zizheng Pan, Yefei He, Gholamreza Haffari, and Bohan Zhuang. 2024. MiniCache: KV Cache Compression in Depth Dimension for Large Language Models. In *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (Eds.), Vol. 37. Curran Associates, Inc., 139997–140031. doi:10.52202/079017-4443

[95] Jun Liu, Zhenglun Kong, Changdi Yang, Fan Yang, Tianqi Li, Peiyan Dong, Joannah Nanjekye, Hao Tang, Geng Yuan, Wei Niu, et al. 2025. Rcr-router: Efficient role-aware context routing for multi-agent llm systems with structured memory. *arXiv preprint arXiv:2508.04903* (2025).

[96] Lei Liu, Xiaoyan Yang, Yue Shen, Binbin Hu, Zhiqiang Zhang, Jinjie Gu, and Guannan Zhang. 2023. Think-in-memory: Recalling and post-thinking enable llms with long-term memory. *arXiv preprint arXiv:2311.08719* (2023).

[97] 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 (2024), 157–173.

[98] Ruiheng Liu, Jinyu Zhang, Yanqi Song, Yu Zhang, and Bailong Yang. 2025. Filling memory gaps: Enhancing continual semantic parsing via sql syntax variance-guided llms without real data replay. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 24641–24649.

[99] Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. 2025. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. *arXiv preprint arXiv:2508.09736* (2025).

[100] Junfeng Lu and Yueyan Li. 2025. Dynamic Affective Memory Management for Personalized LLM Agents. *arXiv preprint arXiv:2510.27418* (2025).

[101] Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2025. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *IEEE Transactions on Audio, Speech and Language Processing* (2025).

[102] 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* (2024).

[103] Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121* (2024).

[104] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. 2023. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems* 36 (2023), 46212–46244.

[105] Vasilije Markovic, Lazar Obradovic, Laszlo Hajdu, and Jovan Pavlovic. 2025. Optimizing the Interface Between Knowledge Graphs and LLMs for Complex Reasoning. *arXiv preprint arXiv:2505.24478* (2025).

[106] James L McClelland, Bruce L McNaughton, and Randall C O'Reilly. 1995. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review* 102, 3 (1995), 419.

[107] Kai Mei, Xi Zhu, Wujiang Xu, Wenyue Hua, Mingyu Jin, Zelong Li, Shuyuan Xu, Ruosong Ye, Yingqiang Ge, and Yongfeng Zhang. 2024. Aios: Llm agent operating system. *arXiv preprint arXiv:2403.16971* (2024).

[108] Justin J Miller. 2013. Graph database applications and concepts with Neo4j. In *Proceedings of the southern association for information systems conference, Atlanta, GA, USA*, Vol. 2324. 141–147.

[109] Atsuyuki Miyai, Zaiying Zhao, Kazuki Egashira, Atsuki Sato, Tatsumi Sunada, Shota Onohara, Hiromasa Yamanishi, Mashiro Toyooka, Kunato Nishina, Ryoma Maeda, et al. 2025. WebChoreArena: Evaluating Web Browsing Agents on Realistic Tedious Web Tasks. *arXiv preprint arXiv:2506.01952* (2025).

[110] 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* (2023).

[111] Amirkeivan Mohtashami and Martin Jaggi. 2023. Random-access infinite context length for transformers. *Advances in Neural Information Processing Systems* 36 (2023), 54567–54585.

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

[113] Maria Paz Oliva, Adriana Correia, Ivan Vankov, and Viktor Botev. 2025. The illusion of a perfect metric: Why evaluating AI's words is harder than it looks. *arXiv preprint arXiv:2508.13816* (2025).

[114] Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. 2023. MemGPT: Towards LLMs as Operating Systems. (2023).

[115] Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Xufang Luo, Hao Cheng, Dongsheng Li, Yuqing Yang, Chin-Yew Lin, H Vicky Zhao, Lili Qiu, et al. 2025. Secom: On memory construction and retrieval for personalized conversational agents. In *The Thirteenth International Conference on Learning Representations*.

[116] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*. 1–22.

[117] Mathis Pink, Qinyuan Wu, Vy Ai Vo, Javier Turek, Jianing Mu, Alexander Huth, and Mariya Toneva. 2025. Position: Episodic Memory is the Missing Piece for Long-Term LLM Agents. *arXiv preprint arXiv:2502.06975* (2025).

[118] David Premack and Guy Woodruff. 1978. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences* 1, 4 (1978), 515–526.

[119] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, et al. 2024. Chatdev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 15174–15186.

[120] Hongjin Qian, Peitian Zhang, Zheng Liu, Kelong Mao, and Zhicheng Dou. 2024. Memorag: Moving towards next-gen rag via memory-inspired knowledge discovery. *arXiv preprint arXiv:2409.05591* 1 (2024).

[121] Ruichen Qiu, Jiajun Tan, Jiayue Pu, Honglin Wang, Xiao-Shan Gao, and Fei Sun. 2025. A Survey on Unlearning in Large Language Models. *arXiv preprint arXiv:2510.25117* (2025).

[122] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PmLR, 8748–8763.

[123] Vignav Ramesh and Kenneth Li. 2025. Communicating Activations Between Language Model Agents. *arXiv preprint arXiv:2501.14082* (2025).

[124] 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* (2025).

[125] Alireza Rezazadeh, Zichao Li, Ange Lou, Yuying Zhao, Wei Wei, and Yujia Bao. 2025. Collaborative Memory: Multi-User Memory Sharing in LLM Agents with Dynamic Access Control. *arXiv preprint arXiv:2505.18279* (2025).

[126] Alireza Rezazadeh, Zichao Li, Wei Wei, and Yujia Bao. 2024. From isolated conversations to hierarchical schemas: Dynamic tree memory representation for llms. *arXiv preprint arXiv:2410.14052* (2024).

[127] Samarth Sarin, Lovepreet Singh, Bhaskarjit Sarmah, and Dhagash Mehta. 2025. Memoria: A Scalable Agentic Memory Framework for Personalized Conversational AI. *arXiv preprint arXiv:2512.12686* (2025).

[128] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems* 36 (2023), 68539–68551.

[129] Junxiao Shen, John J Dudley, and Per Ola Kristensson. 2024. Encode-Store-Retrieve: Augmenting Human Memory through Language-Encoded Egocentric Perception. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 923–931.

[130] Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A Smith, and Chiyuan Zhang. 2024. Muse: Machine unlearning six-way evaluation for language models. *arXiv preprint arXiv:2407.06460* (2024).

[131] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems* 36 (2023), 8634–8652.

[132] Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2020. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768* (2020).

[133] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. 2024. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18221–18232.

[134] Xin Song, Zhikai Xue, Guoxiu He, Jiawei Liu, and Wei Lu. 2025. Interweaving Memories of a Siamese Large Language Model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 25155–25163.

[135] George Sperling. 1960. The information available in brief visual presentations. *Psychological monographs: General and applied* 74, 11 (1960), 1.

[136] Larry R Squire. 2004. Memory systems of the brain: a brief history and current perspective. *Neurobiology of learning and memory* 82, 3 (2004), 171–177.

[137] Haoran Sun and Shaoning Zeng. 2025. Hierarchical memory for high-efficiency long-term reasoning in llm agents. *arXiv preprint arXiv:2507.22925* (2025).

[138] Zequn Sun, Qingheng Zhang, Wei Hu, Chengming Wang, Muhao Chen, Farahnaz Akrami, and Chengkai Li. 2020. A benchmarking study of embedding-based entity alignment for knowledge graphs. *arXiv preprint arXiv:2003.07743* (2020).

[139] Richard S Sutton, Doina Precup, and Satinder Singh. 1999. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence* 112, 1-2 (1999), 181–211.

[140] Mirac Suzgun, Mert Yuksekgonul, Federico Bianchi, Dan Jurafsky, and James Zou. 2025. Dynamic cheatsheet: Test-time learning with adaptive memory. *arXiv preprint arXiv:2504.07952* (2025).

[141] Haoran Tan, Zeyu Zhang, Chen Ma, Xu Chen, Quanyu Dai, and Zhenhua Dong. 2025. MemBench: Towards More Comprehensive Evaluation on the Memory of LLM-based Agents. *arXiv preprint arXiv:2506.21605* (2025).

[142] Mohammad Tavakoli, Alireza Salemi, Carrie Ye, Mohamed Abdalla, Hamed Zamani, and J Ross Mitchell. 2025. Beyond a Million Tokens: Benchmarking and Enhancing Long-Term Memory in LLMs. *arXiv preprint arXiv:2510.27246* (2025).

[143] Shulin Tian, Ruiqi Wang, Hongming Guo, Penghao Wu, Yuhao Dong, Xiuying Wang, Jingkang Yang, Hao Zhang, Hongyuan Zhu, and Ziwei Liu. 2025. Ego-R1: Chain-of-Tool-Thought for Ultra-Long Egocentric Video Reasoning. *arXiv preprint arXiv:2506.13654* (2025).

[144] Luanbo Wan and Weizhi Ma. 2025. StoryBench: A Dynamic Benchmark for Evaluating Long-Term Memory with Multi Turns. *arXiv preprint arXiv:2506.13356* (2025).

[145] Bing Wang, Xinnian Liang, Jian Yang, Hui Huang, Shuangzhi Wu, Peihao Wu, Lu Lu, Zejun Ma, and Zhoujun Li. 2023. Enhancing large language model with self-controlled memory framework. *arXiv preprint arXiv:2304.13343* (2023).

[146] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291* (2023).

[147] Hengyi Wang, Haizhou Shi, Shiwei Tan, Weiyi Qin, Wenyuan Wang, Tunyu Zhang, Akshay Nambi, Tanuja Ganu, and Hao Wang. 2025. Multimodal needle in a haystack: Benchmarking long-context capability of multimodal large language models. 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)*. 3221–3241.

[148] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science* 18, 6 (2024), 186345.

[149] Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Advances in Neural Information Processing Systems* 37 (2024), 53764–53797.

[150] Qingyue Wang, Yanhe Fu, Yanan Cao, Shuai Wang, Zhiliang Tian, and Liang Ding. 2025. Recursively summarizing enables long-term dialogue memory in large language models. *Neurocomputing* 639 (2025), 130193.

[151] Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. 2025. Lvbench: An extreme long video understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 22958–22967.

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

[153] Yu Wang, Yifan Gao, Xiusi Chen, Haoming Jiang, Shiyang Li, Jingfeng Yang, Qingyu Yin, Zheng Li, Xian Li, Bing Yin, et al. 2024. Memoryllm: Towards self-updatable large language models. *arXiv preprint arXiv:2402.04624* (2024).

[154] Yu Wang, Dmitry Krotov, Yuanzhe Hu, Yifan Gao, Wangchunshu Zhou, Julian McAuley, Dan Gutfreund, Rogerio Feris, and Zexue He. 2025. M+: Extending MemoryLLM with Scalable Long-Term Memory. *arXiv preprint arXiv:2502.00592* (2025).

[155] Yu Wang, Ryuichi Takanobu, Zhiqi Liang, Yuzhen Mao, Yuanzhe Hu, Julian McAuley, and Xiaojian Wu. 2025. Mem-{\alpha}: Learning Memory Construction via Reinforcement Learning. *arXiv preprint arXiv:2509.25911* (2025).

[156] Yutong Wang, Jiali Zeng, Xuebo Liu, Derek F Wong, Fandong Meng, Jie Zhou, and Min Zhang. 2024. Delta: An online document-level translation agent based on multi-level memory. *arXiv preprint arXiv:2410.08143* (2024).

[157] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, et al. 2024. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[158] Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. 2024. Agent workflow memory. *arXiv preprint arXiv:2409.07429* (2024).

[159] Aman Kassahun Wassie, Mahdi Molaei, and Yasmin Moslem. 2024. Domain-specific translation with open-source large language models: Resource-oriented analysis. *arXiv preprint arXiv:2412.05862* (2024).

[160] Benjue Weng. 2024. Navigating the landscape of large language models: A comprehensive review and analysis of paradigms and fine-tuning strategies. *arXiv preprint arXiv:2404.09022* (2024).

[161] Lilian Weng. 2023. LLM-Powered Autonomous Agents. https://lilianweng.github.io/posts/2023-06-23-agent/. Online blog post.

[162] Schaun Wheeler and Olivier Jeunen. 2025. Procedural memory is not all you need: Bridging cognitive gaps in llm-based agents. In *Adjunct Proceedings of the 33rd ACM Conference on User Modeling, Adaptation and Personalization*. 360–364.

[163] Matthew A Wilson and Bruce L McNaughton. 1994. Reactivation of hippocampal ensemble memories during sleep. *Science* 265, 5172 (1994), 676–679.

[164] 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* (2024).

[165] Haoyi Wu and Kewei Tu. 2024. Layer-Condensed KV Cache for Efficient Inference of Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 11175–11188. doi:10.18653/v1/2024.acl-long.602

[166] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. 2024. Autogen: Enabling next-gen LLM applications via multi-agent conversations. In *First Conference on Language Modeling*.

[167] Shiguang Wu, Yaqing Wang, and Quanming Yao. 2025. Dense Communication between Language Models. *arXiv preprint arXiv:2505.12741* (2025).

[168] Yaxiong Wu, Sheng Liang, Chen Zhang, Yichao Wang, Yongyue Zhang, Huifeng Guo, Ruiming Tang, and Yong Liu. 2025. From human memory to ai memory: A survey on memory mechanisms in the era of llms. *arXiv preprint arXiv:2504.15965* (2025).

[169] Haomiao Xiong, Zongxin Yang, Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Jiawen Zhu, and Huchuan Lu. 2025. Streaming video understanding and multi-round interaction with memory-enhanced knowledge. *arXiv preprint arXiv:2501.13468* (2025).

[170] Haoran Xu, Jiacong Hu, Ke Zhang, Lei Yu, Yuxin Tang, Xinyuan Song, Yiqun Duan, Lynn Ai, and Bill Shi. 2025. SEDM: Scalable Self-Evolving Distributed Memory for Agents. *arXiv preprint arXiv:2509.09498* (2025).

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

[172] Wei Xu, Gang Luo, Weiyu Meng, Xiaobing Zhai, Keli Zheng, Ji Wu, Yanrong Li, Abao Xing, Junrong Li, Zhifan Li, et al. 2025. MRAgent: an LLM-based automated agent for causal knowledge discovery in disease via Mendelian randomization. *Briefings in Bioinformatics* 26, 2 (2025), bbaf140.

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

[174] Jingkang Yang, Shuai Liu, Hongming Guo, Yuhao Dong, Xiamengwei Zhang, Sicheng Zhang, Pengyun Wang, Zitang Zhou, Binzhu Xie, Ziyue Wang, et al. 2025. Egolife: Towards egocentric life assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 28885–28900.

[175] 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*. 2369–2380.

[176] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

[177] Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. 2025. WorldMM: Dynamic Multimodal Memory Agent for Long Video Reasoning. *arXiv preprint arXiv:2512.02425* (2025).

[178] Shuo Yu, Mingyue Cheng, Daoyu Wang, Qi Liu, Zirui Liu, Ze Guo, and Xiaoyu Tao. 2025. MemWeaver: A Hierarchical Memory from Textual Interactive Behaviors for Personalized Generation. *arXiv preprint arXiv:2510.07713* (2025).

[179] Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, et al. 2022. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598* (2022).

[180] Ruihong Zeng, Jinyuan Fang, Siwei Liu, and Zaiqiao Meng. 2024. On the structural memory of llm agents. *arXiv preprint arXiv:2412.15266* (2024).

[181] Guibin Zhang, Muxin Fu, Guancheng Wan, Miao Yu, Kun Wang, and Shuicheng Yan. 2025. G-Memory: Tracing Hierarchical Memory for Multi-Agent Systems. *arXiv preprint arXiv:2506.07398* (2025).

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

[183] Heng Zhang, Yuling Shi, Xiaodong Gu, Haochen You, Zijian Zhang, Lubin Gan, Yilei Yuan, and Jin Huang. 2025. D3MAS: Decompose, Deduce, and Distribute for Enhanced Knowledge Sharing in Multi-Agent Systems. *arXiv preprint arXiv:2510.10585* (2025).

[184] Wanlong Zhang, Tongfei Liu, Yong Su, and Shuang Zhu. 2025. Breaking the Trust Paradox: Machine Unlearning via Neighbor-Collaborative Forgetting and Regret Updating. In *International Conference on Intelligent Computing*. Springer, 158–169.

[185] Wenqi Zhang, Ke Tang, Hai Wu, Mengna Wang, Yongliang Shen, Guiyang Hou, Zeqi Tan, Peng Li, Yueting Zhuang, and Weiming Lu. 2024. Agent-pro: Learning to evolve via policy-level reflection and optimization. *arXiv preprint arXiv:2402.17574* (2024).

[186] Xinliang Frederick Zhang, Nick Beauchamp, and Lu Wang. 2025. Prime: Large language model personalization with cognitive dual-memory and personalized thought process. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 33695–33724.

[187] Yi Zhang, Zhongyang Yu, Wanqi Jiang, Yufeng Shen, and Jin Li. 2023. Long-term memory for large language models through topic-based vector database. In *2023 International Conference on Asian Language Processing (IALP)*. IEEE, 258–264.

[188] Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2025. A survey on the memory mechanism of large language model-based agents. *ACM Transactions on Information Systems* 43, 6 (2025), 1–47.

[189] Zeyu Zhang, Quanyu Dai, Luyu Chen, Zeren Jiang, Rui Li, Jieming Zhu, Xu Chen, Yi Xie, Zhenhua Dong, and Ji-Rong Wen. 2024. Memsim: A bayesian simulator for evaluating memory of llm-based personal assistants. *arXiv preprint arXiv:2409.20163* (2024).

[190] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 19632–19642.

[191] Canzhe Zhao, Yanjie Ze, Jing Dong, Baoxiang Wang, and Shuai Li. 2023. DPMAC: Differentially private communication for cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2308.09902* (2023).

[192] Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. 2025. Do llms recognize your preferences? evaluating personalized preference following in llms. *arXiv preprint arXiv:2502.09597* (2025).

[193] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223* 1, 2 (2023).

[194] Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. 2023. Synapse: Trajectory-as-exemplar prompting with memory for computer control. *arXiv preprint arXiv:2306.07863* (2023).

[195] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372* (2024).

[196] Yujia Zheng, Zhuokai Zhao, Zijian Li, Yaqi Xie, Mingze Gao, Lizhu Zhang, and Kun Zhang. 2025. Thought Communication in Multiagent Collaboration. *arXiv preprint arXiv:2510.20733* (2025).

[197] 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*, Vol. 38. 19724–19731.

[198] Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, and Jun Wang. 2025. Memento: Fine-tuning LLM Agents without Fine-tuning LLMs. arXiv:2508.16153 [cs.LG] https://arxiv.org/abs/2508.16153

[199] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, et al. 2025. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 13691–13701.

[200] Wangchunshu Zhou, Yuchen Eleanor Jiang, Peng Cui, Tiannan Wang, Zhenxin Xiao, Yifan Hou, Ryan Cotterell, and Mrinmaya Sachan. 2023. Recurrentgpt: Interactive generation of (arbitrarily) long text. *arXiv preprint arXiv:2305.13304* (2023).

[201] Jiaru Zou, Xiyuan Yang, Ruizhong Qiu, Gaotang Li, Katherine Tieu, Pan Lu, Ke Shen, Hanghang Tong, Yejin Choi, Jingrui He, et al. 2025. Latent Collaboration in Multi-Agent Systems. *arXiv preprint arXiv:2511.20639* (2025).

[202] Jialong Zuo, Yongtai Deng, Lingdong Kong, Jingkang Yang, Rui Jin, Yiwei Zhang, Nong Sang, Liang Pan, Ziwei Liu, and Changxin Gao. 2025. VideoLucy: Deep Memory Backtracking for Long Video Understanding. *arXiv preprint arXiv:2510.12422* (2025).