MEMORY NAVIGATES, THE RECORD DECIDES: AUDITABLE EVIDENCE FOR MILLION-TOKEN AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Even when a million-token history fits in the context window, an agent cannot attend to all of it at every step, so it answers from compressed memory. Memory systems that answer from model-written summaries and facts can use outdated or contradictory information and cannot tell when evidence is missing. We observe that compiled memory can afford to be lossy if it only points to source lines that the agent re-reads, so we use it only to navigate to evidence and let the re-read record decide the answer. NAVMEM, a training-free compiler, builds a multi-resolution continuity view of the history and a dynamics view that tracks updates, conflicts, and event order. Code stamps each item with the address of its source lines, so every item is auditable, and the answer protocol asks the agent to check each claim against the re-read lines and to abstain when they do not support it. When every system answers with Qwen3.5-122B, NAVMEM scores 0.715 macro on BEAM's 1M tier, 0.100 above the strongest baseline, Mem0 (paired 95% CI $[0.076, 0.125]$), and 0.755 accuracy on LoCoMo, 0.066 above LIGHT ($[0.051, 0.083]$).

1 INTRODUCTION

An agent that serves one user across many sessions accumulates about a million tokens of history. Even when that history fits in the context window, the model cannot attend to all of it at every step. Rebuilding a coding agent's context costs an estimated 5,000–20,000 tokens per session (Malo & Qiu, 2026), and keeping the raw history in context dilutes attention and degrades reasoning at out-of-distribution lengths (Zhang et al., 2026; Zhou et al., 2026b). Agents therefore *compile* their history into a working memory they can carry. We argue that the central failure of compilation is *unverifiable remembering*. Once a compiled memory stands in for the events that produced it, the agent can answer confidently from outdated or contradictory information, and it cannot tell that it lacks evidence, so it does not abstain (Figure 1).

We restrict compiled memory to finding evidence. Every answer should be re-derivable from the raw transcript (the record), and the agent should abstain when it is not; we call this requirement **grounding**. Because compiled memory is not meant to supply answers, it can afford to be lossy. It must still provide **continuity**, compressing a history too long to carry verbatim into a state the agent keeps across sessions, and **dynamics**, keeping superseded and current values reachable as facts are updated or contradicted and as events occur in order.

Consolidating memory systems answer from irreversible summaries that the model itself wrote (Section 2). Retrieval systems keep the record but usually pass retrieved passages to the reader without checking claims against them. Few designs combine answer-time checking against the record with the compiled state an agent must carry across sessions. Both needs can be met if compiled memory only finds evidence and each item keeps its source address. NAVMEM (navigation memory) builds on this idea without training and compiles a million-token transcript into a **continuity view**, a multi-resolution summary of the history, and a **dynamics view** that keeps facts and events with deterministic update and conflict edges. Its answer protocol instructs the agent to re-read the cited lines and to abstain without support.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

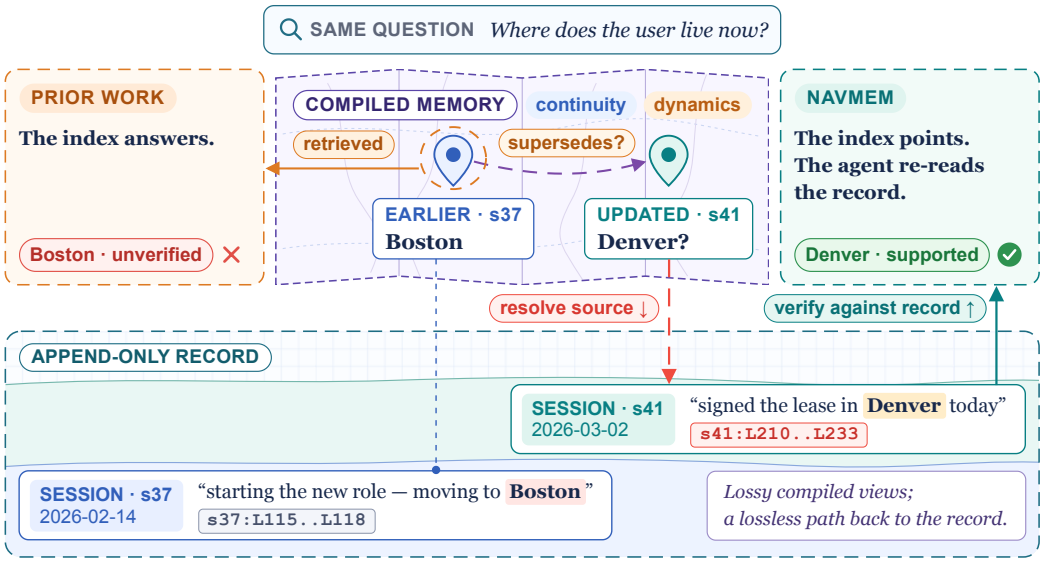


Figure 1: **Compiled memory points to the record.** Memory holds Boston (s37) and the newer Denver (s41), linked by an unverified supersession edge. Prior work answers with the retrieved stale value. In NAVMEM, the edge only leads to the newer candidate. The candidate's address resolves to exact lines in the append-only record, which the agent is instructed to re-read before accepting the claim.

Contributions.

- Memory systems that answer from compressed summaries and facts can use outdated or contradictory information and cannot tell when evidence is missing. We observe that compiled memory can afford to be lossy if it only points to source lines that the agent re-reads (Proposition 1), so memory should locate evidence and leave the answer to the re-read record.
- NAVMEM compiles the history into a continuity view and a dynamics view and reads the raw transcript through an evidence interface (Section 3). Code writes every item's source address, so the compiler can drop content without losing where a claim came from. Deterministic rules link updated and conflicting facts, so a current value can be traced to its source. The protocol instructs the agent to re-read the addressed lines, take current values from the record, and abstain without support, so when the agent complies, the record decides the answer.
- When every system answers with Qwen3.5-122B, NAVMEM scores 0.715 macro on BEAM's 1M tier against 0.615 for Mem0, the strongest baseline (paired 95% CI of the gap $[0.076, 0.125]$), and stays ahead under a second judge ($+0.080$, $[0.053, 0.107]$). On LoCoMo it reaches 0.755 against 0.689 for LIGHT ($[0.051, 0.083]$; Section 4.2). Consistent with the insight, the full system reaches 0.803 on information extraction and 0.761 on knowledge update, against 0.419 and 0.243 with the continuity view alone (Section 4.3).

2 RELATED WORK

Continuity versus authority. Long-context models process longer inputs through recurrence, compression, or attention sinks (Dai et al., 2019; Rae et al., 2020; Xiao et al., 2024), yet still underuse evidence placed mid-context (Liu et al., 2024). Agent memory systems instead carry state across sessions. MemGPT pages between an active window and external tiers (Packer et al., 2023). MemoryBank, Generative Agents, MemoryOS, LIGHT, Mem0, and Zep add forgetting, reflection, compression, or fact updates (Zhong et al., 2024; Park et al., 2023; Kang et al., 2025; Tavakoli et al., 2026; Chhikara et al., 2025; Rasmussen et al., 2025). A-MEM evolves linked notes as observations

arrive (Xu et al., 2025). RL-trained policies learn what to keep (Zhang et al., 2026; Zhou et al., 2026b; Yu et al., 2026). These systems increase what an agent can carry, but their compressed items can answer without their sources. Retrieval augmentation keeps the record and fetches from it (Lewis et al., 2020; Borgeaud et al., 2022), through summary trees, gists, and graph indexes (Sarthi et al., 2024; Chen et al., 2023; Lee et al., 2024; Gutiérrez et al., 2024), multi-granularity routing (Xu et al., 2026a), or reflective evidence-gap tracking (Du et al., 2025). They choose what to fetch but do not check each claim in the answer against the fetched lines, and the retrieval itself is often imprecise. Embedding retrieval reaches only 0.09 precision on same-domain beliefs (Flynt, 2026), and grep often beats vectors on LongMemEval (Wu et al., 2025; Sen et al., 2026).

Provenance and evidence lifecycle. Several recent systems link memory to its sources. MemWeaver pairs graph and experience memories with a passage memory for traceability (Ye et al., 2026), retrieving turn-level passages through entities instead of line pointers. TierMem links summaries to an immutable raw tier and escalates when a learned sufficiency router predicts a miss (Zhu et al., 2026). MemIR types raw evidence, retrieval cues, and truth-bearing claims (Jin et al., 2026), MemoryLACE derives merge, supersession, and contradiction relations over atomic memories (Yacoubi et al., 2026), Hindsight separates facts, experiences, summaries, and beliefs (Latimer et al., 2026), and PROJECTMEM projects summaries from an authoritative log (Malo & Qiu, 2026). Agent Zero Memory answers under a citation lock (Wu & Zhu, 2026), and SodaMem reads span-verified facts with supersede and contradict edges through a planner–reader loop (Wan et al., 2026). The closest concurrent works are NapMem, which learns by reinforcement learning to navigate a provenance-linked, multi-granularity memory pyramid (Xu et al., 2026b), and AuthMem-Bench, which shows that consolidation can keep a claim while erasing the source constraints on its use (Zhan et al., 2026). NAVMEM uses compiled items only to locate evidence, instructs the agent to trace every accepted assertion to addressed raw lines, derives update and conflict edges in code, and trains no component.

Dynamics and evaluation. Recent benchmarks also score updates, conflicts, and forgetting (Hu et al., 2026; Tan et al., 2025; Uddin et al., 2026; Chen et al., 2025). Diagnostic studies single out updates. Models struggle with contradictions even under oracle evidence (Wang et al., 2026b) and can retrieve an update yet accept the stale premise (Chao et al., 2026). Under dynamic, static, and conditional conflicts they miss, misrank, or misuse memories (Tao et al., 2026), and they violate remembered user corrections despite retrieval (Zhou et al., 2026a). EvoMemBench finds that no memory representation wins across task families (Wang et al., 2026c). We evaluate on BEAM's 1M tier, which probes update, contradiction, and ordering directly, and use LoCoMo (Maharana et al., 2024) as an external check. Appendix A compares design choices and BEAM comparability across systems. We focus on single-agent cross-session memory and exclude multi-agent memory and parametric consolidation (Behrouz et al., 2026).

3 NAVIGATION MEMORY OVER AN IMMUTABLE RECORD

NAVMEM uses compiled items only as pointers into the transcript. It keeps two source-linked views because continuity needs a compressed history while dynamics needs superseded and current values to stay reachable, and it serves grounding with tools that read the raw transcript (Figure 2).

Problem formulation. An agent's experience stream $S = (u_1, \ldots, u_T)$ is the append-only sequence of transcript lines from its sessions, too long for the model to attend to in full at each step. An address $(s, [i, j])$ names lines $i..j$ of session s, $S^{(s)}[i..j]$ denotes them re-parsed as messages, and $\mathsf{Ref}(S)$ is the set of valid addresses. Given a question q, the agent outputs an answer a or abstains ($\perp$), and each assertion in a should be re-derivable from a span of S. NAVMEM works in two steps. As sessions arrive, it incrementally compiles compact views whose items point to their source lines (Definition 1), and at answer time the agent is instructed to re-read those lines before deciding. Because LLMs compile the views, runs over the same S can differ, so ω indexes a run.

Definition 1 (Source-addressed compiler run)**.** *A compiler run ω maps the stream to a continuity view, a dynamics view, and a resolution map, and every compiled item m carries an address $\mathrm{ref}(m) = (s, [i, j]) \in \mathsf{Ref}(S)$ that resolves to the original lines:*

$$\mathcal{C}_\omega(S) = (\mathcal{V}^\omega_{\mathrm{con}}, \mathcal{V}^\omega_{\mathrm{dyn}}, \Pi_S), \qquad \Pi_S(\mathrm{ref}(m)) = S^{(s)}[i..j] \quad \forall\, m \in \mathcal{V}^\omega_{\mathrm{con}} \cup \mathcal{V}^\omega_{\mathrm{dyn}}. \tag{1}$$

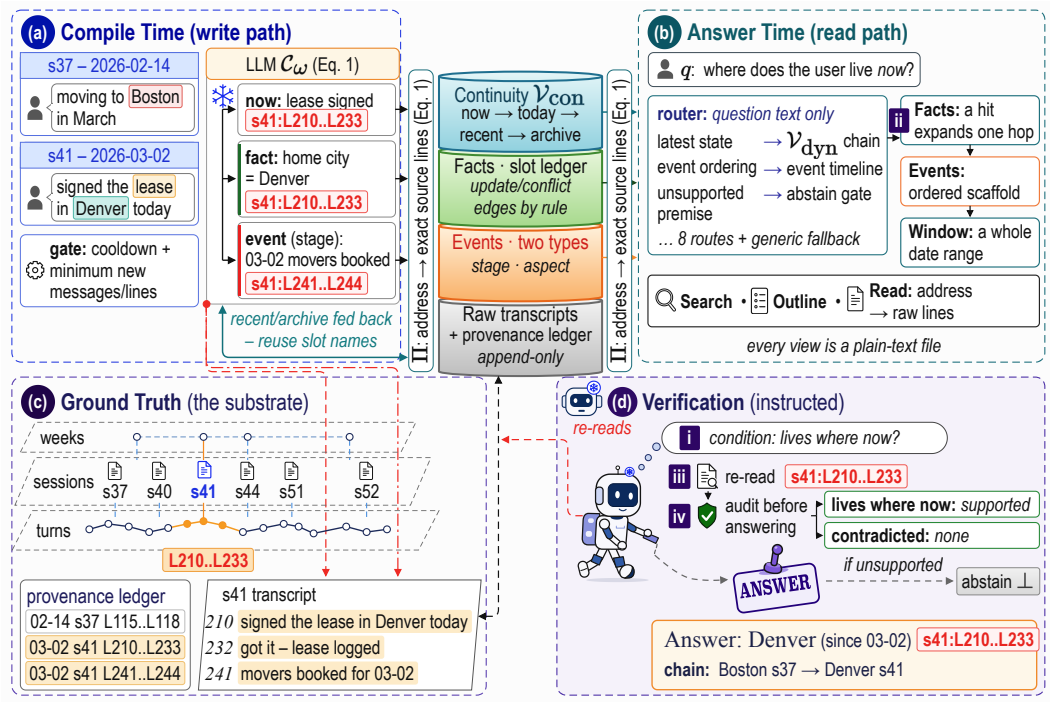


Figure 2: **The NAVMEM architecture on a worked example** (illustrative). **(a) Compile time:** an LLM proposes continuity, fact, and event content, and code stamps each item with its source address. **Center:** the continuity view $\mathcal{V}_{\text{con}}$ and the rule-linked fact–event ledger sit over the raw substrate and resolution map Π. **(b) Answer time:** a router selects a retrieval strategy, and the evidence interface reads the views and raw substrate. **(c) Ground truth:** the append-only record keeps turn-, session-, and week-scale evidence. **(d) Protocol:** the agent is instructed to re-read addressed lines, emit an evidence audit, and abstain when support is missing (Algorithm 1).

The model proposes each item's payload, and code stamps $\text{ref}(m)$, so a compressed item can lose content but not its source address, which Π_S re-reads. Eq. (1) holds for every run. The transcript is append-only by storage policy.

At answer time, the agent's trace τ, its sequence of tool calls for one question, gathers candidates from the views and re-reads their addresses, yielding raw spans E_τ. The protocol instructs the agent to label each condition of q as supported, contradicted, or missing in E_τ and to abstain unless every assertion is supported (Section 3.5, formalized in Appendix B as Eq. (2) and Algorithm 1).

Proposition 1 (Views affect answers only through evidence)**.** *If the agent's decision is a function $a = D(q, E_\tau)$ of the question and the re-read evidence alone, then any two view states that lead to the same E_τ yield the same answer. The views can change which evidence is found and have no other effect on the answer.*

Proof. View payloads are not arguments of D; they affect a only through which addresses are resolved, and those are fixed by E_τ. □

The protocol asks the agent to satisfy this premise, but the agent runs on a prompted model and nothing enforces it. Section 4.4 measures which lines it re-reads and cites, without testing the premise itself. Appendix D states when acceptance and abstention are semantically correct and which properties the edge rules guarantee.

3.1 SUBSTRATE AND PROVENANCE

The substrate consists of append-only session transcripts and a provenance ledger. Each time the compiler captures a new slice of experience, it records in the provenance ledger which session and lines the slice came from. Every item in every compiled view carries its source address in the form

$\langle$*sid*:$L\ell_0..L\ell_1\rangle$, which writes out $\mathrm{ref}(m)$ from Definition 1. A reader tool resolves such a reference by re-reading and re-parsing the referenced lines from the on-disk transcript, with no cache or pre-embedded copy in between.

3.2 Compiling for Continuity

The continuity view has four time layers, now, today, recent, and archive, from newest to oldest, and older layers are compressed further. A capture runs only after a cooldown and a minimum number of new human messages and transcript lines. Each accepted capture makes one LLM call that writes a one-line now summary. The compiler rewrites the continuity state instead of appending to it. When now exceeds a size threshold or a time budget, a compression call folds it into the day's today layer. A consolidation call then merges today into recent and archive, so that recent holds the state of a recent time window and archive holds older state, with no growing log of weekly entries. Mechanical checks keep truncated or malformed rewrites from replacing the existing state (Appendix D). Code strips source attributions from model output and re-attaches them from the provenance ledger, so the model can compress content but cannot corrupt addresses. The rewrite compresses only along the time axis and leaves current values and contradictions to the dynamics view.

3.3 Compiling for Dynamics

The dynamics view $\mathcal{V}_{\mathrm{dyn}}$ consists of two append-only ledgers, one for facts and one for events, whose entries record text, a source address, a timestamp, and a slot key. Events also carry an order key. The slot key names the quantity a fact assigns, such as the user's home city in Figure 2. An LLM extracts facts and events from the raw transcript (models in Appendix F). Its prompt withholds edge decisions entirely, so every candidate it returns is an unlinked fact or event with a source range. Deterministic rules then derive the edges between facts. A supersession edge links a fact to the next later fact in the same slot family (shared slot-key tokens or a common prefix) whose normalized value differs. A conflict edge links a first-person denial ("I haven't deployed it") to a completion claim ("deployment is done") when the two share an action family and enough topic overlap. Each rule is a structural check, and none uses an embedding-similarity threshold. The same extracted facts therefore rebuild the same rule-derived edges; an LLM pass adds 75 of the 5,536 stored edges (1.4%). Appendix C gives the idealized rules (Eqs. (3)–(5)). Retrieval over this view is lexical, and each hit expands one hop along its supersession and conflict edges while keeping its source address. The edges only route the search. The agent is instructed to use the value re-read from the raw transcript.

3.4 Evidence Access

The evidence interface comprises ledger queries and four tools that read the transcript. A transcript reader resolves any source address to re-parsed raw lines, a window reader returns the lines in a date range regardless of topic, an outline tool returns the transcript's structure, and a search tool matches terms lexically. When an external embedding endpoint is configured, the search tool can also fuse its scores. A person or an agent can read and search every view directly. The prompt also carries the continuity archive and the routed candidates with their source addresses.

3.5 Query-Time Routing and the Verification Protocol

The agent runs a multi-turn tool loop over the views. A deterministic keyword classifier maps the question text alone to a retrieval route (latest state, temporal, event ordering, summary, arithmetic, instruction, unsupported premise, or recommendation) or to a generic fallback. As guidance in the prompt, the route sets which view to consult first and which evidence gate applies, that is, what evidence an answer needs. For example, a latest-state answer must confirm that no later value exists, and for an unsupported-premise question the agent abstains without support. All tools remain available, so routing reorders the search without truncating it.

The protocol instructs the agent to decompose the question into required conditions and gather candidates from the routed view. Before relying on a candidate, the agent resolves the candidate's source address, re-reads the raw lines, and marks each condition supported, contradicted, or missing. It then emits an evidence audit listing what it read and how it decided, and gives the final answer.

Nothing checks whether a cited span supports the answer (Appendix H), so we call this verification *prompt-instructed* and measure how often the agent follows it.

4 EXPERIMENTS

RQ1 asks whether a memory that uses compiled views only to locate evidence outperforms the evaluated baselines (Section 4.2); **RQ2**, how individual components and pairs of components compare with the full system (Section 4.3); and **RQ3**, how closely the agent follows the protocol and whether citing gold sources goes with higher accuracy (Section 4.4). A scaling study varies the answering model and the context length (Section 4.2).

4.1 SETUP

Benchmarks and metrics. Our main benchmark is BEAM (Tavakoli et al., 2026), whose synthetic multi-session conversations stay narratively coherent at controlled lengths. We report its 1M-token tier (35 conversations, 700 questions, two per ability in each conversation across ten abilities). The headline score is the ability-balanced macro of rubric-nugget judgments, with event ordering scored by the Mem0-faithful convention (Eqs. (6)–(7), Appendix E). Each conversation is replayed into the compiler as dated sessions, and the reported runs use the offline prebuilt ledger (Appendix H). LoCoMo (Maharana et al., 2024) (10 conversations, 1,986 questions) is the external check (single-round protocol, Appendix F), scored by LLM-judge accuracy. Questions within a conversation share one memory build, so the conversation is the unit of replication, and we report 95% conversation-cluster bootstrap intervals for NAVMEM (10,000 resamples, fixed seed). Paired intervals against baselines apply each resample to both systems. The main NAVMEM row is a single first-pass run (Appendix G).

Compared systems. All systems share the answering model and the judge and are evaluated end to end. Baselines keep their own readers and retrieval workflows, and NAVMEM answers under its protocol, with at most 30 tool turns, a keyword router, an evidence gate, three retry types (format, no-tool, route quality), and a lexical search tool with optional embedding fusion (Section 3.4). The main comparison is therefore between whole systems, and Section 4.3 fixes the router and the answering model to attribute the gain among components. Appendix F lists each system's build, retrieval budget, tools, retries, and prompts. The systems are (i) **No Memory**, the answering model without memory; (ii) **NativeRAG**, passive top-k retrieval over the raw transcript (Lewis et al., 2020); (iii) **Mem0** (Chhikara et al., 2025), a production persistent-memory layer; (iv) **LIGHT** (Tavakoli et al., 2026), the benchmark's own memory baseline; (v) **A-MEM** (Xu et al., 2025), agentic memory with evolving note links; (vi) **MemoryOS** (Kang et al., 2025), a hierarchical personal-memory system;[1] and (vii) **NAVMEM** (ours).

Models and development samples. The main answering model is a self-hosted Qwen3.5-122B (GPTQ-Int4, 256K context), and GPT-5.5 judges every run. The NAVMEM agent runs on the Claude Agent SDK inside Anthropic's sandbox runtime, which blocks the agent's own network access and keeps the gold answers out of its reach. The scaling study (Appendix G) adds Qwen3.6-35B-A3B (FP8) and GPT-5.6; the GPT-5.6 multi-method blocks (Tables 13 and 15) use the first three conversations. The self-hosted models run on 16 NVIDIA A100 80GB GPUs. We tuned the keyword router's patterns and the retry ladder on conversations 1–10 of the 1M tier with a different answering model (Claude Haiku 4.5). These conversations are also scored in the main grid, and on conversations 11–35, held out from this tuning, NAVMEM still leads Mem0 by 0.103 ($[0.072, 0.134]$; Appendix G). No model was trained, and all rules stay fixed across every reported cell.

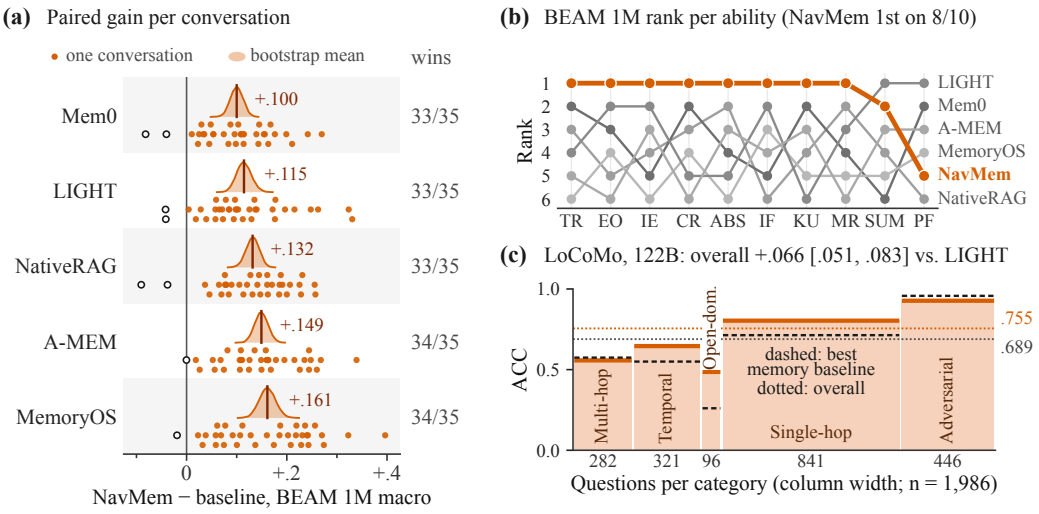


Figure 3: NAVMEM leads every baseline on BEAM 1M and LoCoMo (Qwen3.5-122B). **(a)** Per-conversation macro differences (dots) and bootstrap distributions of the mean difference (shaded; 10,000 resamples; Table 10); No Memory is off scale. **(b)** Rank of six systems on each ability (abbreviations and scores in Table 2). **(c)** LoCoMo judge accuracy by category (Table 5); column width proportional to question count, dashed lines mark the best memory baseline, dotted lines overall accuracy.

Table 1: Main results with Qwen3.5-122B. BEAM 1M scores are ten-ability macros. Δ is NAVMEM minus the baseline with a paired 95% conversation-cluster CI, and wins count the conversations, of 35, whose macro favors NAVMEM (Table 10). LoCoMo scores are question-pooled accuracy. MemoryOS was not run on LoCoMo.

	BEAM 1M			LoCoMo
Method	Macro	Δ [95% CI]	Wins	Accuracy
No Memory	0.141	+0.573 [0.543, 0.603]	35/35	0.225
NativeRAG	0.583	+0.132 [0.106, 0.157]	33/35	0.579
Mem0	0.615	+0.100 [0.076, 0.125]	33/35	0.687
MemoryOS	0.553	+0.161 [0.130, 0.193]	34/35	–
LIGHT	0.600	+0.115 [0.087, 0.144]	33/35	0.689
A-MEM	0.565	+0.149 [0.123, 0.176]	34/35	0.534
NAVMEM	**0.715**	–	–	**0.755**

4.2 MAIN RESULTS (RQ1)

On the BEAM 1M tier under Qwen3.5-122B (RQ1), full NAVMEM outperforms every evaluated baseline (Table 1 and Figure 3a), reaching 0.715 ± 0.013 macro (95% conversation-cluster CI $[0.690, 0.740]$), 0.100 above the strongest external baseline, Mem0 (0.615), and 0.132 above passive retrieval (NativeRAG, 0.583); Qwen3.5-122B without memory scores 0.141. With conversations resampled jointly, the paired 95% CI of the difference over Mem0 is $[0.076, 0.125]$, and NAVMEM leads on 33 of 35 conversations. Under GLM-5.2, a judge from another model family, the gain is $+0.080$ ($[0.053, 0.107]$; 31 of 35; Table 3). Table 10 gives the other baselines. Among the six systems ranked in Figure 3b and Table 2, NAVMEM is first on eight of the ten abilities (one tied), with its largest margins over Mem0 in information extraction (0.803 vs. 0.600), instruction following (0.802

[1]For MemoryOS under Qwen3.5-122B, the memory store was built by Qwen3.6-35B-A3B and reused for answering, and answers left empty at 8,192 tokens were regenerated at 16,384 (19 stayed empty); no other system approached that limit. On LoCoMo, LIGHT and A-MEM under Qwen3.5-122B likewise answer over memories built by Qwen3.6-35B-A3B.

Table 2: Per-ability BEAM 1M scores with Qwen3.5-122B. ABS=Abstention, CR=Contradiction Resolution, EO=Event Ordering, IE=Information Extraction, IF=Instruction Following, KU=Knowledge Update, MR=Multi-session Reasoning, PF=Preference Following, SUM=Summarization, and TR=Temporal Reasoning. The best score in each column is bold. Table 13 adds the other answering models.

Method	ABS	CR	EO	IE	IF	KU	MR	PF	SUM	TR	Macro
No Memory	**1.000**	0.000	0.251	0.000	0.071	0.000	0.013	0.055	0.000	0.025	0.141
NativeRAG	0.707	0.637	0.437	0.600	0.586	0.614	**0.619**	0.745	0.464	0.418	0.583
Mem0	0.650	0.738	0.458	0.600	0.613	0.732	0.563	0.807	0.443	0.542	0.615
MemoryOS	0.643	0.605	0.453	0.565	0.683	0.557	0.524	0.768	0.446	0.288	0.553
LIGHT	0.650	0.543	0.493	0.709	0.743	0.536	0.565	**0.829**	**0.572**	0.361	0.600
A-MEM	0.654	0.527	0.416	0.626	0.677	0.625	0.503	0.801	0.485	0.339	0.565
NAVMEM	0.789	**0.823**	**0.590**	**0.803**	**0.802**	**0.761**	**0.619**	0.760	0.517	**0.682**	**0.715**

Table 3: Second judge on BEAM 1M. The same first-pass NAVMEM and Mem0 answers (Qwen3.5-122B, 35 conversations) are scored by GPT-5.5 and by GLM-5.2. Paired conversation-cluster bootstrap and sign-flip test as in Table 10; every sign-flip $p < 10^{-3}$. "Both complete" keeps the 494 questions (one on summarization) with both full answers judged.

Judge	NAVMEM	Mem0	Δ [95% paired CI]	Wins
All paired questions (700)				
GPT-5.5	0.715	0.615	+0.100 [0.076, 0.125]	33/35
GLM-5.2	0.692	0.611	+0.080 [0.053, 0.107]	31/35
Both complete (494)				
GPT-5.5	0.722	0.619	+0.104 [0.078, 0.141]	32/35
GLM-5.2	0.702	0.622	+0.080 [0.054, 0.110]	29/35

Table 4: Answering-model scaling on the BEAM 1M tier (macro), three systems. Qwen3.6-35B-A3B and Qwen3.5-122B cover all 35 conversations and are the only columns compared with each other. The GPT-5.6 column reports only NAVMEM. Bold is the best system in each column.

System	Qwen3.6-35B-A3B	Qwen3.5-122B	GPT-5.6
No Memory	0.138	0.141	–
NativeRAG	0.557	0.583	–
NAVMEM	**0.677**	**0.715**	**0.755**

vs. 0.613), temporal reasoning (0.682 vs. 0.542), abstention (0.789 vs. 0.650), and event ordering (0.590 vs. 0.458).

On LoCoMo, an external check, NAVMEM retrieves once and answers without tool calls, as the baselines do, and reaches 0.755 question-pooled accuracy (95% CI [0.743, 0.769]) against 0.689 for LIGHT and 0.687 for Mem0 (Figure 3c), a paired difference of +0.066 over LIGHT ([0.051, 0.083]). On the four factual categories (1,540 questions), NAVMEM reaches 0.706 against 0.619 for LIGHT and 0.614 for Mem0, with gains over the strongest baseline in temporal (0.646 vs. 0.550), open-domain (0.486 vs. 0.260), and single-hop questions (0.804 vs. 0.714; Table 5). The lead holds under Qwen3.6-35B-A3B (0.750 vs. 0.714 for Mem0, the strongest baseline there; +0.036, [0.019, 0.053]).

Context and model scaling. With either Qwen model answering on all 35 conversations, NAVMEM exceeds NativeRAG and the model without memory (Table 4). Across the three answering models its score falls by at most 0.016 from 100K (20 conversations) to 1M (35) and drops at 10M (10 conversations; 0.575 with Qwen3.5-122B), mainly on multi-session and temporal reasoning (Table 14).

4.3 COMPONENT ABLATION (RQ2)

The full configuration, which uses the views to locate transcript lines, exceeds evidence-only access by 0.086 on BEAM (paired 95% CI [0.062, 0.110]; RQ2), a gap that also reflects its prompt, candidate table, and retries. On BEAM, removing any one component lowers the score on the same questions (Table 6). The full system exceeds evidence with continuity by 0.046 ([0.017, 0.076]), evidence with dynamics by 0.043 ([0.007, 0.078]; 21 conversations), and continuity with dynamics by 0.129 ([0.107, 0.149]). On LoCoMo, removing evidence or dynamics costs 0.189 and 0.056, whereas removing continuity costs 0.004 ([−0.006, 0.014]), consistent with its far shorter conversations. Each

Table 5: LoCoMo judge accuracy (ACC, GPT-5.5 judge; 10 conversations, 1,986 questions). Per-category columns use Qwen3.5-122B; the last two columns give question-pooled ACC for both answering models. NAVMEM runs the same single-round protocol under both models: one forced retrieval, then the answer, over memory built by GLM-5-Turbo. Under Qwen3.5-122B, LIGHT and A-MEM answer over memories built by Qwen3.6-35B-A3B. Best score per column in bold.

	Qwen3.5-122B, per category					Overall	
Method	Multi-hop	Temporal	Open-dom.	Single-hop	Adv.	35B-A3B	122B
No Memory	0.000	0.000	0.000	0.000	**1.000**	0.225	0.225
NativeRAG	0.245	0.500	0.169	0.566	0.957	0.565	0.579
Mem0	**0.574**	0.550	0.260	0.693	0.937	0.714	0.687
LIGHT	0.557	0.545	0.227	0.714	0.929	0.698	0.689
A-MEM	0.276	0.466	0.157	0.482	0.925	0.545	0.534
NAVMEM	0.557	**0.646**	**0.486**	**0.804**	0.927	**0.750**	**0.755**

Table 6: Component ablation (Qwen3.5-122B). Evidence: raw-transcript retrieval; Continuity, Dynamics: the view as injected text. Δ: paired difference from the full system on the same questions (Table 17); Table 16 gives all ten abilities. †Full-system macro plus the paired Δ over the questions on which the agent ran its tool loop (Appendix G).

	BEAM 1M		LoCoMo	
Configuration	Macro	Δ	ACC	Δ
No Memory	0.141	−0.573	0.225	−0.531
Evidence only	0.629	−0.086	0.666	−0.090
Continuity only	0.564	−0.151	0.354	−0.401
Dynamics only	0.480	−0.235	0.521	−0.235
Evidence + Continuity†	0.668	−0.046	0.700	−0.056
Evidence + Dynamics†	0.672	−0.043	0.752	−0.004
Continuity + Dynamics	0.586	−0.129	0.566	−0.189
Full NAVMEM	**0.715**	–	**0.755**	–

pair there also outscores both of its components. Used alone as answer context, each compiled view scores below evidence-only access (0.564 and 0.480 vs. 0.629). The continuity view alone keeps the gist of a conversation but not its exact values, as expected of a lossy view. It scores 0.646 on event ordering and 0.592 on summarization, above the full system, but 0.419 on information extraction and 0.243 on knowledge update, against 0.803 and 0.761 (Table 16). Adding dynamics raises knowledge update to 0.671 over continuity; as 84.2% of facts have no supersession edge, these arms do not isolate the edges' effect.

4.4 MECHANISM DIAGNOSTICS (RQ3)

RQ3 tests whether the agent cites the evidence the protocol asks for. A rule-based parser counts a raw read when the agent reads transcript lines by address or searches inside one session, and a citation when the answer names an address $(s, [i, j])$. A gold source counts as cited when a cited range contains it. Across the 630 logged episodes on BEAM 1M (Table 12), it re-reads the transcript in 98.9% of episodes, cites source lines in 82.9%, and emits the evidence audit in 72.9%. It cites a gold source in 58.2% of the 555 episodes that have one, and on multi-state questions (n=204) 51.5% of episodes cite every reference group. These citation rates come from saved 1,200-character prefixes and are lower bounds. On the 617 episodes whose full response is recoverable, source-line citation rises from 0.828 to 0.882 (Appendix G). Episodes that cite a gold source score 0.120 higher (95% CI $[0.071, 0.168]$; observational).

5 CONCLUSION

We presented NAVMEM, a training-free memory that addresses unverifiable remembering by using lossy views only to locate evidence and instructing the agent to ground each claim in the re-read transcript. When every system answers with Qwen3.5-122B, it scores 0.715 macro on BEAM's 1M tier against 0.615 for Mem0 and 0.755 accuracy on LoCoMo against 0.689 for LIGHT. The agent carries out this verification as the protocol instructs, and the system does not yet enforce it. Episodes that cite a gold source score 0.120 higher, so the next step is to enforce it in the system.

Reproducibility Statement

Section 4.1 and Appendix F specify the answering models, the judge, and every baseline configuration. Appendix B gives the answer protocol, Appendix C the edge rules, and Appendix E the metrics and bootstrap procedure. The archived per-question records cover BEAM, LoCoMo, the scaling study, and the component ablations; we will release them with the analysis scripts that regenerate the reported scores, the paired intervals (Table 10), and the diagnostics (Table 12).

AI Use Statement

Large language models are components of the systems studied here (the answering models, the fact and event extractors, and the judge); Section 4.1 and Appendix F specify each. In preparing this work, we used generative AI tools to translate the authors' revisions between Chinese and English, to give feedback on the methodology and experiments through simulated reviews, and to check the propositions in Appendix D and revise their proofs. We have not used generative AI tools to propose hypotheses, formulate the propositions, develop the conceptual framework, implement the methods, or interpret the results, and generating synthetic data, cleaning datasets, and qualitative data analysis are not applicable to this work. Additionally, we used generative AI tools to draft and edit text, create figures, write analysis and plotting code, search the literature, and check bibliographic metadata. We have reviewed all AI-assisted work, checking every reported number against the archived per-question records and every citation against its source, and we take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Ali Behrouz, Farnoosh Hashemi, Adel Javanmard, and Vahab Mirrokni. Language models need sleep: Learning to self-modify and consolidate memories, 2026. URL https://arxiv.org/abs/2606.03979.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 2206–2240. PMLR, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Hanxiang Chao, Yihan Bai, Rui Sheng, Tianle Li, and Yushi Sun. STALE: Can LLM agents know when their memories are no longer valid? *arXiv preprint arXiv:2605.06527*, 2026. URL https://arxiv.org/abs/2605.06527.

Ding Chen, Simin Niu, Kehang Li, Peng Liu, Xiangping Zheng, Bo Tang, Xinchi Li, Feiyu Xiong, and Zhiyu Li. HaluMem: Evaluating hallucinations in memory systems of agents, 2025. URL https://arxiv.org/abs/2511.03506.

Howard Chen, Ramakanth Pasunuru, Jason Weston, and Asli Celikyilmaz. Walking down the memory maze: Beyond context limit through interactive reading, 2023. URL https://arxiv.org/abs/2310.05029.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. In *Frontiers in Artificial Intelligence and Applications: ECAI 2025*, pp. 2993–3000, 2025. doi: 10.3233/FAIA251160. URL https://ebooks.iospress.nl/doi/10.3233/FAIA251160.

Zeyang Cui, Jiannong Cao, Zhiyuan Wen, Bo Yuan, Junlan Feng, and Shengyuan Chen. EdgeMem: LLM-free agent memory construction and retrieval via evidence-preserving multi-anchor hypergraph, 2026. URL https://arxiv.org/abs/2609.05553.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2978–2988, 2019. doi: 10.18653/v1/P19-1285. URL https://aclanthology.org/P19-1285/.

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

Jeffrey Flynt. Structured belief state and the first precision-aware benchmark for LLM memory retrieval, 2026. URL https://arxiv.org/abs/2605.11325. Version 4.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, 2023. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569, 2024. doi: 10.52202/079017-1902. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/6ddc001d07ca4f319af96a3024f6dbd1-Paper-Conference.pdf.

Jun He and Deying Yu. Stored is not supported: Typed provenance and assertion guardrails for persistent AI agents, 2026. URL https://arxiv.org/abs/2609.02127.

Hao-Lun Hsu, Nikki Lijing Kuang, Boyi Liu, Zhewei Yao, and Yuxiong He. Organize then retrieve: Hierarchical memory navigation for efficient agents, 2026. URL https://arxiv.org/abs/2606.11680.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=DT7JyQC3MR.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, Senjie Jin, Jiejun Tan, Yanbin Yin, Jiongnan Liu, Zeyu Zhang, Zhongxiang Sun, Yutao Zhu, Hao Sun, Boci Peng, Zhenrong Cheng, Xuanbo Fan, Jiaxin Guo, Xinlei Yu, Zhenhong Zhou, Zewen Hu, Jiahao Huo, Junhao Wang, Yuwei Niu, Yu Wang, Zhenfei Yin, Xiaobin Hu, Yue Liao, Qiankun Li, Kun Wang, Wangchunshu Zhou, Yixin Liu, Dawei Cheng, Qi Zhang, Tao Gui, Shirui Pan, Yan Zhang, Philip Torr, Zhicheng Dou, Ji-Rong Wen, Xuanjing Huang, Yu-Gang Jiang, and Shuicheng Yan. Memory in the age of AI agents, 2025. URL https://arxiv.org/abs/2512.13564.

HydraDB. Long-term memory for AI agents at million-token scale: BEAM 1M. https://benchmarks.hydradb.com/beam-1m, 2026. Vendor-reported result; accessed 2026-09-25.

Suozhao Ji, Baodong Wu, Zehao Wang, Lei Xia, Qingping Li, Ruisong Wang, Wenbo Ding, Zhenhua Zhu, Boxun Li, Guohao Dai, and Yu Wang. Infini memory: Maintainable topic documents for long-term LLM agent memory, 2026. URL https://arxiv.org/abs/2606.10677.

Zhengda Jin, Bingbing Wang, Jing Li, Ruifeng Xu, and Min Zhang. Mitigating provenance-role collapse in long-term agents via typed memory representation, 2026. URL https://arxiv.org/abs/2605.25869.

Resham Joshi. Eywa: Provenance-grounded long-term memory for AI agents, 2026. URL https://arxiv.org/abs/2605.30771.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. Memory OS of AI agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 25961–25970, 2025. doi: 10.18653/v1/2025.emnlp-main.1318. URL https://aclanthology.org/2025.emnlp-main.1318/.

Christopher Latimer, Nicolò Boschi, Andrew Neeser, Chris Bartholomew, Gaurav Srivastava, Xuan Wang, and Naren Ramakrishnan. Hindsight: Structured agent memory that retains, recalls, and reflects. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 275–285, 2026. doi: 10.18653/v1/2026.acl-demo.27. URL https://aclanthology.org/2026.acl-demo.27/.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. A human-inspired reading agent with gist memory of very long contexts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 26396–26415. PMLR, 2024. URL https://proceedings.mlr.press/v235/lee24c.html.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13851–13870, 2024. doi: 10.18653/v1/2024.acl-long.747. URL https://aclanthology.org/2024.acl-long.747/.

Ripon Chandra Malo and Tong Qiu. PROJECTMEM: A local-first, event-sourced memory and judgment layer for AI coding agents, 2026. URL https://arxiv.org/abs/2606.12329.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems, 2023. URL https://arxiv.org/abs/2310.08560.

Joon Sung Park, Joseph C. O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–22, 2023. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. MemoRAG: Boosting long context processing with global memory-enhanced retrieval augmentation. In *Proceedings of the ACM on Web Conference 2025*, pp. 2366–2377, 2025. doi: 10.1145/3696410.3714805. URL https://dl.acm.org/doi/10.1145/3696410.3714805.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SylKikSYDH.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: A temporal knowledge graph architecture for agent memory, 2025. URL https://arxiv.org/abs/2501.13956.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GN921JHCRw.

Sahil Sen, Akhil Kasturi, Elias Lumer, Anmol Gulati, and Vamse Kumar Subbiah. Is grep all you need? how agent harnesses reshape agentic search, 2026. URL https://arxiv.org/abs/2605.15184.

Quentin Spencer. Ground truth first: A longitudinal evaluation instrument for agent memory, and the tenure crossover in memory-architecture rankings. *arXiv preprint arXiv:2607.21962*, 2026. URL https://arxiv.org/abs/2607.21962.

Haoran Tan, Zeyu Zhang, Chen Ma, Xu Chen, Quanyu Dai, and Zhenhua Dong. MemBench: Towards more comprehensive evaluation on the memory of LLM-based agents. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19336–19352. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.989. URL https://aclanthology.org/2025.findings-acl.989/.

Bo Tang, Yang Zhang, Guomian Zhuang, Wenqiang Wei, Gaoyang Zheng, Lindong Xie, Yanchao Tan, Feiyu Xiong, Qingyu Yang, Edward Chung, and Zhiyu Li. From memory to skills: Evidence-grounded co-evolution governance for long-horizon LLM agents. *arXiv preprint arXiv:2607.16621*, 2026. URL https://arxiv.org/abs/2607.16621.

Zhen Tao, Jinxiang Zhao, Peng Liu, Dinghao Xi, Yanfang Chen, Wei Xu, and Zhiyu Li. MemConflict: Evaluating long-term memory systems under memory conflicts. *arXiv preprint arXiv:2605.20926*, 2026. URL https://arxiv.org/abs/2605.20926.

Mohammad Tavakoli, Alireza Salemi, Carrie Ye, Mohamed Abdalla, Hamed Zamani, and J. Ross Mitchell. Beyond a million tokens: Benchmarking and enhancing long-term memory in LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=y59hf5lrMn.

Md Nayem Uddin, Kumar Shubham, Eduardo Blanco, Chitta Baral, and Gengyu Wang. From recall to forgetting: Benchmarking long-term memory for personalized agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 26814–26841. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1337. URL https://aclanthology.org/2026.findings-acl.1337/.

Vectorize. Agent memory benchmark: BEAM leaderboard. https://agentmemorybenchmark.ai/dataset/beam, 2026. Vendor-run leaderboard; accessed 2026-09-25.

Fengrong Wan, Chengcan Wu, and Ningtao Lyu. SodaMem: Evidence-grounded temporal graph memory for LLM agents, 2026. URL https://arxiv.org/abs/2608.08055.

Changjian Wang, Rongzhen Li, Weili Guan, Shuming Shi, Quan Lu, and Ning Jiang. CueMem: Cue-guided context reconstruction for long-term conversational memory, 2026a. URL https://arxiv.org/abs/2609.12354.

Wenxuan Wang, Haoyu Sun, Fukuan Hou, Mingyang Song, Weinan Zhang, Yu Cheng, and Yang Yang. SubtleMemory: A benchmark for fine-grained relational memory discrimination in long-horizon AI agents, 2026b. URL https://arxiv.org/abs/2606.05761.

Yuyao Wang, Zhongjian Zhang, Mo Chi, Kaichi Yu, Yuhan Li, Miao Peng, Bing Tong, Chen Zhang, Yan Zhou, and Jia Li. EvoMemBench: Benchmarking agent memory from a self-evolving perspective. *arXiv preprint arXiv:2605.18421*, 2026c. URL https://arxiv.org/abs/2605.18421.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=pZiyCaVuti.

Ming Wu and Pengyuan Zhu. Agent Zero Memory: Provenance-aware long-term memory for LLM agents, 2026. URL https://arxiv.org/abs/2608.29606.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, pp. 21875–21895, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5e5fd18f863cbe6d8ae392a93fd271c9-Paper-Conference.pdf.

Derong Xu, Yi Wen, Pengyue Jia, Yingyi Zhang, Wenlin Zhang, Yichao Wang, Huifeng Guo, Ruiming Tang, Xiangyu Zhao, Enhong Chen, and Tong Xu. From single to multi-granularity: Toward long-term memory association and selection of conversational agents. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=i2yIvZARnG.

Guodong Xu. Governed persistent memory: Source-bound state semantics and fail-closed release for long-horizon agents, 2026. URL https://arxiv.org/abs/2608.12476.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-Mem: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, pp. 17577–17604, 2025. doi: 10.52202/085713-0593. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/19909c36f51abc4856b4560aff3d36d6-Paper-Conference.pdf.

Yue Xu, Yutao Sun, Yihao Liu, Mengyu Zhou, Jiayi Qiao, Lu Ma, Kai Tang, Wenjie Wang, Xiaoxi Jiang, and Guanjun Jiang. From passive retrieval to active memory navigation: Learning to use memory as a structured action space, 2026b. URL https://arxiv.org/abs/2607.05794.

Meriem Yacoubi, Pia Schmidt, Nenad Petrovic, Ahmed Frikha, Martin Kirchhoff, and Alois Knoll. MemoryLACE: Memory lifecycle-aware consolidation and evidence retrieval. *arXiv preprint arXiv:2609.03201*, 2026. URL https://arxiv.org/abs/2609.03201.

Juexiang Ye, Xue Li, Xinyu Yang, Chengkai Huang, Lanshun Nie, Lina Yao, and Dechen Zhan. MemWeaver: Weaving hybrid memories for traceable long-horizon agentic reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 12928–12956, 2026. doi: 10.18653/v1/2026.findings-acl.630. URL https://aclanthology.org/2026.findings-acl.630/.

Yi Yu, Liuyi Yao, Yuexiang Xie, Qingquan Tan, Jiaqi Feng, Yaliang Li, and Libing Wu. Agentic memory: Learning unified long-term and short-term memory management for large language model agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21457–21483, 2026. doi: 10.18653/v1/2026.acl-long.981. URL https://aclanthology.org/2026.acl-long.981/.

Qiuyang Zhan, Rui Zhang, Sheng Guo, Lepeng Zhao, and Zhuotao Liu. When memory becomes authority: Benchmarking authority collapse at the memory consolidation boundary, 2026. URL https://arxiv.org/abs/2608.01679.

Yuxiang Zhang, Jiangming Shu, Ye Ma, Xueyuan Lin, Shangxi Wu, and Jitao Sang. Memory as action: Autonomous context curation for long-horizon agentic tasks. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 19149–19164, 2026. doi: 10.18653/v1/2026.findings-acl.956. URL https://aclanthology.org/2026.findings-acl.956/.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19724–19731, 2024. doi: 10.1609/aaai.v38i17.29946. URL https://ojs.aaai.org/index.php/AAAI/article/view/29946.

Yujun Zhou, Kehan Guo, Haomin Zhuang, Xiangqi Wang, Yue Huang, Zhenwen Liang, Pin-Yu Chen, Tian Gao, Nuno Moniz, Nitesh V. Chawla, and Xiangliang Zhang. Getting better at working with you: Compiling user corrections into runtime enforcement for coding agents, 2026a. URL https://arxiv.org/abs/2606.13174.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Bryan Kian Hsiang Low, and Paul Pu Liang. MEM1: Learning to synergize memory and reasoning for efficient long-horizon agents. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=XY8AaxDSLb.

Qiming Zhu, Shunian Chen, Rui Yu, Zhehao Wu, and Benyou Wang. From lossy to verified: A provenance-aware tiered memory for agents, 2026. URL https://arxiv.org/abs/2602.17913. ICLR 2026 Workshop on Memory for LLM-Based Agentic Systems.

A Contribution Delta (Related Work)

Table 7 summarizes, for the nearest systems discussed in Section 2, what each system's own description claims and what its public paper lets us verify. Cont. denotes a continuity view across sessions, Prov. that items resolve to source evidence, Dyn. explicit update or conflict structure, Read an answer-time protocol that re-reads source evidence, and Train whether any component is trained. Earlier reading agents already navigate summary trees or gists back to raw pages for single-document QA (Chen et al., 2023; Lee et al., 2024), MemoRAG drafts clues from a global memory to retrieve raw passages (Qian et al., 2025), and Self-RAG and ALCE critique or cite retrieved support (Asai et al., 2024; Gao et al., 2023). NavMem carries this navigate-then-check pattern into cross-session memory with updates and conflicts. Other concurrent systems keep immutable evidence authoritative and treat extraction as an index (Joshi, 2026), use memories as cues to reconstruct source-turn context (Wang et al., 2026a), or retrieve raw turns without an LLM (Cui et al., 2026). Plain-text topic documents (Ji et al., 2026) and RL-trained file-system navigation (Hsu et al., 2026) target maintainability and token efficiency, and a field survey calls trustworthy memory a foundational deployment requirement (Hu et al., 2025). Concurrent governance layers decide which stored records may support an outgoing claim (Xu, 2026; He & Yu, 2026).

Beam comparability. Our Beam-1M numbers use Qwen3.5-122B to answer and GPT-5.5 to judge, so they are not directly comparable to Beam scores reported elsewhere. The Beam/LIGHT paper uses its own aggregate evaluation, and MemIR and MemoryLACE report only the 100K tier with GPT-4o-mini judges. Vendor-reported Beam-1M accuracies are 73.9% and 79.1% for Hindsight in single-query and retrieval-augmented modes (Vectorize, 2026) and 82% for HydraDB under Hindsight's setup (HydraDB, 2026); they use other answering and grading models and per-question accuracy. We cite these numbers for context only.

B Answer Protocol

This section states the answer-side target that the prompt asks for; the system does not enforce it. We drop ω and S from $\mathcal{V}$ and Π when clear, and $\perp$ denotes abstention. For an answer a, let $\mathrm{Assert}(a)$ be its factual assertions and $\mathrm{cite}(c, a)$ the candidates (view items or raw-substrate hits, each with an address) cited for assertion c. Let C be the labeled records produced when the evidence E_τ of Section 3 is adjudicated: $(c, m, e, \ell) \in C$ means that candidate m was resolved to raw span e and labeled $\ell \in \{\mathsf{sup}, \mathsf{con}, \mathsf{mis}\}$ (supported, contradicted, missing) for claim c. The prompt-targeted admissibility condition and its refusal case are

$$\begin{aligned} \mathrm{Adm}(a; C) \iff\ & a \neq \perp \\ & \wedge\ \forall c \in \mathrm{Assert}(a)\ \exists m \in \mathrm{cite}(c, a) : \\ & \qquad \big(c,\ m,\ \Pi_S(\mathrm{ref}(m)),\ \mathsf{sup}\big) \in C, \end{aligned} \tag{2}$$

$$a_{\mathrm{out}} = \begin{cases} a, & \mathrm{Adm}(a; C), \\ \perp, & \text{otherwise.} \end{cases}$$

The label is indexed by the claim it supports, so a span labeled supported for one condition cannot be cited for another, and an answer without factual assertions is vacuously admissible. A failed or truncated read does not return $\Pi_S(\mathrm{ref}(m))$ and therefore cannot serve as a witness. An answer reporting that the record contradicts a condition c_0 asserts $\neg c_0$ and cites the contradicting span, labeled sup for $\neg c_0$. Equation (1) covers the write side. Under the append-only storage policy, code enforces it for the addresses it stamps, so compiled references resolve to raw spans. Equation (2)

Table 7: Design-contract comparison. Each cell is C/V: C is the system's published claim and V is the property explicitly described in its public paper. $Y/P/N$ denote yes/partial/no; $0/R/M$ denote training-free/router-trained/memory-trained. These are qualitative design attributes, not matched scores.

System	Cont.	Prov.	Dyn.	Read	Train
MemGPT	Y/Y	N/N	N/N	P/P	0/0
Zep[j] (Rasmussen et al., 2025)	Y/Y	Y/P	Y/Y	N/N	0/0
MemWeaver	Y/Y	Y/P	P/P	P/P	0/0
TierMem[a]	Y/Y	Y/Y	P/P	P/P	R/R
MemIR[b]	P/P	Y/Y	P/P	P/P	0/0
MemoryLACE[c]	P/P	Y/Y	Y/Y	N/N	0/0
Hindsight[d]	Y/Y	P/P	Y/Y	N/N	0/0
Agent Zero Memory[e]	Y/Y	Y/Y	Y/P	Y/P	0/0
SodaMem[f]	Y/P	Y/Y	Y/Y	Y/P	0/0
Veracium[g] (Spencer, 2026)	Y/Y	Y/P	Y/Y	P/P	0/0
MSCE[h] (Tang et al., 2026)	Y/Y	Y/P	P/P	P/P	0/0
NAVMEM (Ours)	Y/Y	Y/Y	Y/Y	Y/P[i]	0/0

[a]TierMem answers from summaries on its fast path and escalates to code-linked raw pages ($\sim$1K tokens) when a trained sufficiency router predicts a miss. [b]MemIR's reader sees claims with supporting spans and may abstain; claims, not spans, hold authority. [c]MemoryLACE reports local lifecycle relations and relation-aware retrieval; no claim-level re-read protocol is stated. [d]Hindsight's reflection updates structured beliefs and reports traceable reasoning, but the public description does not establish exact immutable line pointers. [e]Agent Zero cites only opened items, raw or derived, else abstains; pointers resolve to turns or chunks. [f]SodaMem's FactEvents carry ingest-verified literal spans and temporal edges; its reader cites evidence without NAVMEM's labels. [g]Veracium is the layered memory released with the Ground Truth First evaluation instrument; its paper reports no BEAM result. [h]MSCE promotes evidence-linked procedural skills from traces; it also reports LoCoMo, not BEAM. [i]NAVMEM Read is prompt-targeted and measured by the harness; it is not a hard enforcement guarantee. [j]Zep invalidates edges by LLM contradiction checks; its answer context holds facts and summaries, not episodes.

covers the read side, which the prompt targets. Every answer assertion must cite a resolved span labeled supported, and otherwise the instructed output is $\perp$. The condition constrains which assertions survive and says nothing about which inputs the answer depends on, so it neither implies nor is implied by the hypothesis of Proposition 1. Algorithm 1 sketches the tool loop under which the target is prompted.

Algorithm 1 Agent-driven answer loop. The prompt asks for the evidence condition of Eq. (2), and Section 4.4 audits compliance offline.

Require: query q; views $(\mathcal{V}_{\text{con}}, \mathcal{V}_{\text{dyn}})$; reader Π_S; agent $\mathcal{A}$; turn budget $K_{\max} = 30$
$r \leftarrow \textsc{Route}(q)$ ▷ deterministic keyword classifier selects strategy
Prompt $\mathcal{A}$ with route r and evidence verification rules; $M \leftarrow \emptyset$; $E_\tau \leftarrow ()$; answer a unset
for $t = 1 \dots K_{\max}$ **do**
  $\mathcal{A}$ selects action $o_t \in \{\mathsf{view_query}, \mathsf{raw_read}, \mathsf{final_answer}\}$
  if $o_t = \mathsf{view_query}$ **then**
    Add addressed candidates from $\mathcal{V}_{\text{dyn}}$, $\mathcal{V}_{\text{con}}$, or lexical search to M
  else if $o_t = \mathsf{raw_read}$ **then**
    $\mathcal{A}$ selects $m \in M$; on a successful read, append $\Pi_S(\text{ref}(m))$ to E_τ
  else if $o_t = \mathsf{final_answer}$ **then**
    $\mathcal{A}$ emits answer a and evidence audit C over E_τ; **break**
  end if
end for
If the output calls no tool or fails the format or route-quality check, the harness reruns the episode
Log emitted citations and audit for offline analysis
return a (prompt-instructed to equal a_{out} of Eq. (2)); if unset, report budget exhaustion

C EDGE RULES OF THE DYNAMICS VIEW

This section gives the idealized fixed-field rules summarized in Section 3.3, which Appendix D analyzes. The archived builder also depends on record order and filtering, and 75 of its 5,536 stored edges (about 1.4%) come from a separate LLM pass whose proposals pass a local check, so these

equations do not specify every stored edge. For fact f, let $k(f)$ be its slot key, $\mathrm{tok}(k(f))$ its token set, $\mathrm{val}(f)$ its normalized value set, and $\mathrm{ts}(f)$ its timestamp. The slot-family relation is

$$\begin{aligned} f \sim_{\mathrm{sf}} g \iff & \left|\mathrm{tok}(k(f)) \cap \mathrm{tok}(k(g))\right| \geq 2 \\ & \vee\ \exists u \in \mathrm{tok}(k(f)) \cap \mathrm{tok}(k(g)) : |u| \geq 6 \\ & \vee\ \mathrm{pref}(k(f), k(g)), \end{aligned} \tag{3}$$

where pref is a fixed common-prefix test on the two slot keys. With $\mathcal{F}$ denoting the extracted facts, a supersession edge from the prior fact f to the later fact g is present exactly when

$$\begin{aligned} \mathrm{Sup}(f, g) \iff & f \sim_{\mathrm{sf}} g \ \wedge\ \mathrm{ts}(f) < \mathrm{ts}(g) \\ & \wedge\ \mathrm{val}(f) \neq \mathrm{val}(g) \\ & \wedge\ \neg\exists h \in \mathcal{F} :\ h \sim_{\mathrm{sf}} g \\ & \qquad\qquad \wedge\ \mathrm{ts}(f) < \mathrm{ts}(h) < \mathrm{ts}(g). \end{aligned} \tag{4}$$

For conflict detection, let Den and Cmp be the regex classes for first-person historical denials and completion claims, $\mathrm{act}(f)$ the regex-derived action-family set, and $\mathrm{ov}(f, g)$ the overlap count of topic or slot-family terms. Let $\phi(f, g)$ be the hardcoded floor selected by object specificity, with a stronger floor for generic objects such as "task" or "contact" than for a specific object. Then

$$\begin{aligned} \mathrm{Conf}(f, g) \iff & f \neq g \ \wedge\ \begin{bmatrix} \mathrm{Den}(f) \wedge \mathrm{Cmp}(g) \\ \vee\ \mathrm{Den}(g) \wedge \mathrm{Cmp}(f) \end{bmatrix} \\ & \wedge\ \mathrm{act}(f) \cap \mathrm{act}(g) \neq \emptyset \\ & \wedge\ \mathrm{ov}(f, g) \geq \phi(f, g). \end{aligned} \tag{5}$$

Every clause is a structural check (set intersection, regex class membership, timestamp order, exact normalized-value comparison, and discrete token-overlap count thresholds ϕ), and none uses a continuous embedding similarity threshold. The edge set is therefore a function of the extracted facts, and the same facts produce the same edges.

D Formal Properties

This appendix collects what the rules of Section 3 guarantee and where the guarantees stop. None of these statements describes the runtime behavior of the prompted agent. Proposition 1 additionally requires q and D to be fixed and E_τ to contain only the addressed raw-read contents. For a stochastic D it gives identical answer distributions, and individual sampled answers can still differ.

Addresses. In Definition 1, ω indexes one run, including its sampled model outputs, so $\mathcal{C}_\omega$ is a function and a rerun is a different ω. An address $(s, [i, j]) \in \mathsf{Ref}(S)$ names a non-empty line range inside session s, under the reader's index convention. If an item carries several addresses, $\mathrm{ref}(m)$ is read as a set and Eq. (1) elementwise, and $|\mathcal{V}_{\mathrm{con}}|$ is the total length of the continuity files. If S' extends S only by appending lines to sessions or adding sessions, then $\mathsf{Ref}(S) \subseteq \mathsf{Ref}(S')$ and $\Pi_{S'} = \Pi_S$ on $\mathsf{Ref}(S)$, because each $S^{(s)}$ is a prefix of $S'^{(s)}$. Addresses stamped at compile time therefore resolve to the same lines at answer time, and Π_S may render those lines (for example, parse JSON) as long as it does so deterministically. Transcript immutability follows from the append-only policy, and the system does not detect in-place edits. Eq. (1) states only that $\mathrm{ref}(m)$ resolves. The sup label of Eq. (2) concerns whether the resolved span supports the cited assertion c. Each provenance-ledger entry is a tuple $\langle$*date*, *sessionId*, *transcriptPath*, *startLine*, *endLine*$\rangle$, appended by an atomic write-and-rename.

Continuity guard. The consolidation of Section 3.2 keeps the last seven days in recent and folds older state into archive. It is wrapped in three checks. A rewrite shorter than 20% of its input is discarded and re-staged, a consolidation that does not parse into the two sections is retried, and an empty archive never replaces a non-empty one. Each capture emits at most 24 facts and 12 events.

Semantic acceptance and correct refusal. Let $\mathrm{Req}(q)$ be the atomic conditions required by q, and let $e \models_S c$ mean that raw span e entails condition c within the temporal scope of q; $\models_S$ is an ideal annotation relation, not a component of the system. Over all spans $\Pi_S(r)$, $r \in \mathsf{Ref}(S)$, a condition is supported when some span entails it and none entails its negation, contradicted when the reverse

holds, missing when neither holds, and ambiguous when both hold; so $\neg c$ is supported exactly when c is contradicted, and missing or ambiguous exactly when c is. An answer is complete when it asserts c or $\neg c$ for each $c \in \mathrm{Req}(q)$. An answer $a \neq \bot$ is semantically acceptable, $\mathrm{Adm}^{\star}_{S}(a; q)$, when it is complete and every assertion is supported; a refusal is correct when some required condition is missing or ambiguous. A sup label is *span-sound* if $(c, m, e, \mathsf{sup}) \in C$ implies $e \models_S c$, and *condition-sound* if, in addition, no span of S entails $\neg c$.

Proposition 2 (Conditional soundness)**.** *Assume the reader's output for each cited m is $\Pi_S(\mathrm{ref}(m))$. (i) If sup labels are span-sound, every assertion of an answer satisfying Eq.* (2) *is supported or ambiguous; hence, if a required condition is missing, no complete answer satisfies Eq.* (2)*. (ii) If sup labels are condition-sound, every complete answer satisfying Eq.* (2) *satisfies $\mathrm{Adm}^{\star}_{S}$. (iii) When a refusal is correct, no complete answer satisfies $\mathrm{Adm}^{\star}_{S}$.*

Proof. By Eq. (2), each assertion c cites some m with $(c, m, \Pi_S(\mathrm{ref}(m)), \mathsf{sup}) \in C$, and $\Pi_S(\mathrm{ref}(m))$ is one of the spans quantified over. (i) Span-soundness gives a span entailing c, so c is neither missing nor contradicted; for a missing required c_0, both c_0 and $\neg c_0$ are missing, so a complete answer has a missing assertion. (ii) Condition-soundness also excludes a span entailing $\neg c$, so every assertion is supported, and $a \neq \bot$ with completeness gives $\mathrm{Adm}^{\star}_{S}$. (iii) A complete answer asserts c_0 or $\neg c_0$ for a required c_0 that is missing or ambiguous; both are then missing or ambiguous, hence not supported. □

Span-soundness does not give (ii). If one span entails c and another entails $\neg c$, a span-sound sup label on the first passes Eq. (2) although c is ambiguous and a refusal is correct. The protocol has no label for ambiguous conditions. Each witness is a single span. A claim that follows only from several spans, such as an arithmetic combination, needs an evidence-bundle version of Eq. (2), which we leave unstated.

Edge rules. Fix an extracted fact set $\mathcal{F}$ and the tokenizer, value normalizer, regex classes, and thresholds of Eqs. (3)–(5), and let a rebuild recompute all edges from $\mathcal{F}$ without modifying $\mathcal{F}$.

Proposition 3 (Deterministic rebuild and acyclicity)**.** *The idealized supersession and conflict edge sets are uniquely determined by $\mathcal{F}$, rebuilding them is idempotent, and the directed supersession graph is acyclic.*

Proof. Whether an edge joins f and g is a Boolean predicate of the fields of f and g and, through the guard of Eq. (4), of $\mathcal{F}$, computed by fixed functions; a rebuild leaves $\mathcal{F}$ unchanged, so repeating it returns the same sets. Every supersession edge strictly increases the timestamp, so a cycle would give $\mathrm{ts}(f) < \mathrm{ts}(f)$. □

Determinism covers the idealized edges given $\mathcal{F}$. It covers neither the extraction that produces $\mathcal{F}$ nor the archived ledgers, whose builder also depends on record order and candidate inputs (Appendix C).

E Metrics Details

BEAM judging uses the official rubric-nugget protocol, which scores each rubric item in $\{0, 0.5, 1\}$ and averages the items. As in the official protocol, the judge sees each system's complete response, which for NAVMEM includes the evidence audit. With Q_a the questions of ability a and $s(q)$ the rubric-mean of question q, the headline score is the ability-balanced macro over all ten abilities (event ordering enters as the tenth via Eq. (7))

$$\mathrm{macro} \;=\; \frac{1}{10}\sum_{a=1}^{10}\frac{1}{|Q_a|}\sum_{q\in Q_a} s(q). \tag{6}$$

Event ordering matches gold items to the system's ordered answer by LLM-judged equivalence; with $\bar{n}$ the mean per-item match score (each item judged in $\{0, 0.5, 1\}$, missed gold items scoring 0) and τ_b Kendall's τ-b between gold order and the system's order of the matched items,

$$\mathrm{EO} \;=\; \tfrac{1}{2}\big(\bar{n} + \tau_{\mathrm{norm}}\big), \qquad \tau_{\mathrm{norm}} = \tfrac{1}{2}(1 + \tau_b), \tag{7}$$

the *Mem0-faithful* convention.

Table 8 specifies how every reported metric is computed. Event ordering uses the *Mem0-faithful* convention of Eq. (7), which scores missed gold items as zeros inside the per-item match mean and

combines recall and order additively, whereas the BEAM-code `final_score` ($\tau_{\text{norm}} \cdot F_1$) multiplies the order term by the alignment F_1. The two conventions can rank the same runs differently, and all our numbers use the former. Excluding event ordering, the nine-ability macro is 0.728 for NAVMEM and 0.632 for Mem0 (paired difference $+0.096$, 95% CI $[0.071, 0.122]$). The advantage therefore holds outside event ordering.

Table 8: Metric definitions and computation. Each row gives the formula or protocol behind one number reported in the tables.

Metric	Computation
Rubric-nugget $s(q)$	Judge scores each rubric item of question q in $\{0, 0.5, 1\}$; $s(q)$ is the item mean.
Ability score	Mean of $s(q)$ over the ability's questions Q_a.
Macro (Eq. (6))	Unweighted mean of the ten ability scores (event ordering included via Eq. (7)); balances abilities regardless of question counts.
Event ordering (Eq. (7))	Gold items LLM-matched to the system's ordered list; mean of the per-item match mean and normalized Kendall τ-b over matched ranks.
Uncertainty	For NAVMEM: conversation-level mean $\pm$ SEM and 95% cluster-bootstrap CI (10,000 resamples). LoCoMo keeps question-pooled accuracy (ACC) and resamples conversations. Paired baseline intervals apply each resample to both systems (Tables 10 and 11).

F BASELINE CONFIGURATIONS

Table 9 lists how each system was built, how it retrieved, and how it was prompted in the BEAM 1M comparison. Every system answers with Qwen3.5-122B and is judged by GPT-5.5, and each baseline keeps its own memory workflow. Prompts for the non-event-ordering abilities share the harness's `FINAL:` contract, and event-ordering prompts request ordered lists. We wrote the baselines' answer prompts for this harness and kept their official memory-building prompts. Two settings depart from the official defaults for throughput. A-MEM packs two consecutive messages per note (official: one), and MemoryOS uses a retrieval queue of 10 (official: 7). No baseline parameter was searched on conv1–10. Transport retries handle failed requests, and NAVMEM additionally uses format, no-tool, and route-quality retries. The continuity view is built by Qwen3.5-122B, and the fact–event ledger by GLM-4-Air for conversations 1–10 and Qwen3.5-122B for conversations 11–35. We reran LIGHT on conversations 1–8 and 11, which had been run with an earlier version of our LIGHT harness, using the code and settings of the other 26 conversations. On LoCoMo, NAVMEM's views are built by GLM-5-Turbo, and each answer takes a single turn. The harness runs one lexical search over the transcript (4–12 results by route, filtered by the question's target speaker) and injects its lines with a continuity brief (at most 12,000 characters) and ledger hints expanded to their source lines, and the agent calls no tools.

Table 9: Memory and answer settings in the BEAM 1M comparison (Qwen3.5-122B, GPT-5.5 judge, 35 conversations $\times$ 20 probes for every row). Dense baselines use Qwen3-Embedding-0.6B. Input and output budgets are not matched. Retry entries are limits (code defaults). *Departs from the official default. Mem0's ADD-only extraction is the mem0ai default since v2.0.0, which removed the UPDATE/DELETE pass (`https://github.com/mem0ai/mem0/releases/tag/v2.0.0`).

Method	Build / index	Retrieval (budget)	Tools / turns	Retries	Prompts and notes
No Memory	none	none; question only	0 / 1	transport 2	harness answer prompt
NativeRAG	none; 512-word chunks	BM25 (k_1=1.5, b=0.75), top-8	0 / 1	transport 2	harness answer prompt
Mem0	Qwen3.5-122B ADD-only extraction (mem0ai default); local Qdrant	hybrid (mem0ai 2.0.11), top-20, threshold 0.1	0 / 1	answer 5; build 8	official build prompt; 3,000-word ingest batches
MemoryOS	Qwen3.6-35B-A3B; FAISS inner product	dense; 5 sessions, queue 10*, top-20 knowledge	0 / 1	answer 5; build 8	official build prompts; short-term capacity 1
LIGHT	Qwen3.5-122B episodic + scratchpad	dense top-5; scratchpad filter; last 100 pairs	0 / 1	answer 6; build 4	official LIGHT build; reader $\approx$ 14K tokens
A-MEM	Qwen3.5-122B notes with links; Chroma	dense top-5 + linked neighbors ($\leq$ 5)	0 / 1	answer 2; build 8	official note prompts; 2 messages per note*
NAVMEM	Qwen3.5-122B continuity view; ledger by GLM-4-Air (conv. 1–10) or Qwen3.5-122B (11–35)	lexical search over raw lines (limit 8); facts/events top-10, depth 1; optional dense fusion (state not logged)	6 tools / $\leq$ 30	no-tool 1; format 1; route 2	router, evidence gate, protocol; tuned on conv1–10

G ADDITIONAL RESULTS

Extended analysis of the main results. Table 13 gives every ability score. The largest margins over Mem0 are in information extraction ($+0.203$), instruction following ($+0.189$), temporal reasoning ($+0.140$), abstention ($+0.139$), and event ordering ($+0.132$), and the leads in contradiction resolution (0.823 vs. 0.738) and knowledge update (0.761 vs. 0.732, $+0.029$) are smaller. Evidence-only access over the raw transcript yields 0.629, 0.086 below the full system (paired 95% CI $[0.062, 0.110]$; the full system is higher on 30 of 35 conversations; Table 6), and is indistinguishable from Mem0 (paired 95% CI $[-0.012, 0.040]$). The single-component arms share the full system's router, route strategies, and scoring gates. They differ in using arm-specific prompts without candidate tables and in having no route-quality retry. Each pair arm combines two of the three components as in the single arms; the evidence-plus-dynamics arm can also look up the fact and event ledgers. In the two pair arms with evidence, the harness did not run the tool loop on 272 and 377 of the 700 questions (for evidence plus dynamics, on every question of conversations 22–35) and answered them in a single call with only the injected view. We treat these as failed executions: Table 17 compares both arms with the full system on the remaining questions, and Table 6 reports the full-system macro plus that difference (their all-question macros are 0.642 and 0.576). On LoCoMo every arm answers in a single round, as in the main LoCoMo protocol.

Paired comparison and run version. Table 10 resamples the 35 conversations with replacement (10,000 draws, seed 20260925) and applies each draw to both systems, so the interval reflects the per-conversation pairing. The p-value comes from a two-sided sign-flip test over per-conversation macro differences (20,000 permutations). The NAVMEM row is the first-pass run, in which only the 75 probes whose execution failed were rerun. Restricted to conversations 11–35, which played no part in router or retry exploration, the gain over Mem0 is $+0.103$ (paired 95% CI $[0.072, 0.134]$; 23 of 25 conversations). In the Qwen3.6-35B-A3B NAVMEM run, 25 reruns of failed probes (24 event-ordering and 1 instruction-following) used a bounded-retrieval answer policy.

Second judge. To test whether the comparison with Mem0 depends on the judge, we re-scored the archived first-pass NAVMEM and Mem0 answers on BEAM 1M (Qwen3.5-122B, 35 conversations, 700 paired questions) with GLM-5.2 (temperature 0), a judge from another model family, and kept the prompts, rubrics, and per-ability scoring unchanged. The judge sees the complete answer wherever it survives and otherwise the archived first 1,200 characters plus the final line when available (13 NAVMEM and 199 Mem0 answers). Per question, the two judges correlate at 0.909 (Pearson), and GLM-5.2 scores 0.013 lower on average. Under GLM-5.2 the gain over Mem0 is $+0.080$ (paired

Table 10: Paired conversation-cluster bootstrap on BEAM 1M (Qwen3.5-122B, 35 conversations, 700 questions). Δ is NAVMEM macro minus baseline macro; wins count conversations whose per-conversation macro favors NAVMEM. Sign-flip p-values are two-sided and uncorrected for the six comparisons; all stay below 10^{-3} after Holm correction.

Baseline	Δ	95% paired CI	Sign-flip p	Wins
Mem0	$+0.100$	$[0.076, 0.125]$	$< 10^{-3}$	33/35
LIGHT	$+0.115$	$[0.087, 0.144]$	$< 10^{-3}$	33/35
NativeRAG	$+0.132$	$[0.106, 0.157]$	$< 10^{-3}$	33/35
A-MEM	$+0.149$	$[0.123, 0.176]$	$< 10^{-3}$	34/35
MemoryOS	$+0.161$	$[0.130, 0.193]$	$< 10^{-3}$	34/35
No Memory	$+0.573$	$[0.543, 0.603]$	$< 10^{-3}$	35/35

95% CI $[0.053, 0.107]$; 31 of 35 conversations) instead of $+0.100$, and it is also $+0.080$ on the 494 questions where both systems' answers reached the judge complete (Table 3).

On LoCoMo (Table 5), NAVMEM reaches 0.755 question-pooled accuracy under Qwen3.5-122B and 0.750 under Qwen3.6-35B-A3B, above every baseline under each answering model (Table 11). On the four factual categories alone ($1{,}540$ questions), it reaches 0.706 accuracy under Qwen3.5-122B against 0.619 for LIGHT and 0.614 for Mem0, and 0.726 under Qwen3.6-35B-A3B against 0.657 and 0.655. Adversarial questions reward refusal, so No Memory scores 1.000 because it has nothing to answer from, and the memory systems, NAVMEM included, score between 0.925 and 0.957 under Qwen3.5-122B. In the GPT-5.6 block (first three conversations), NAVMEM reaches 0.797 against 0.752 for Mem0 and 0.719 for LIGHT (Table 15). With three conversations we report no interval for this block.

Table 11: Paired conversation-cluster bootstrap on LoCoMo (10 conversations, 1,986 questions; 10,000 resamples, seed 20260925). Δ is NAVMEM ACC minus baseline ACC.

	Qwen3.5-122B		Qwen3.6-35B-A3B	
Baseline	Δ	95% CI	Δ	95% CI
LIGHT	$+0.066$	$[0.051, 0.083]$	$+0.052$	$[0.025, 0.083]$
Mem0	$+0.069$	$[0.049, 0.087]$	$+0.036$	$[0.019, 0.053]$
NativeRAG	$+0.177$	$[0.153, 0.200]$	$+0.185$	$[0.162, 0.209]$
A-MEM	$+0.222$	$[0.204, 0.240]$	$+0.205$	$[0.179, 0.233]$
No Memory	$+0.531$	$[0.509, 0.554]$	$+0.525$	$[0.500, 0.548]$

Extended scaling analysis. On the full 35-conversation BEAM 1M tier, NAVMEM scores 0.677 with Qwen3.6-35B-A3B and 0.715 with Qwen3.5-122B, above NativeRAG (0.557 and 0.583) and the unaugmented baseline (0.138 and 0.141; Table 4). GPT-5.6 reaches 0.755 across all 35 conversations and 0.821 on the three-conversation prefix. From 100K to 1M tokens (Table 14 and Figure 4), NAVMEM shows no consistent decline. Qwen3.5-122B scores 0.711 at 100K, 0.703 at 500K, and 0.715 at 1M, and GPT-5.6 scores 0.768 at 100K, 0.752 at 500K, and 0.755 at 1M. At 10M tokens the score falls (0.575 on Qwen3.5-122B, 0.670 on GPT-5.6), mainly on multi-session and temporal reasoning.

Diagnostic definitions. The diagnostics come from a rule-based parser over the first-pass trajectories. A raw read is a call to the `read` or `within` tool, a source-line citation is a reference of the form `beam-`c`-`s`:L`i`..L`j, and the audit block is the `EVIDENCE AUDIT` section that the protocol requests. A gold source is cited when a cited line interval contains one of the probe's gold source identifiers, and a multi-state probe is fully covered when every non-empty gold group intersects a cited interval. Intervals use $10{,}000$ conversation-cluster percentile bootstrap resamples (seed 20260925), and each resample recomputes the pooled rate as a ratio of summed counts. The archived response keeps only its first $1{,}200$ characters (truncated in 475 of 630 episodes), so the citation, audit, and gold-source rates are lower bounds. On the 617 episodes whose full final response is available (recovered from the session logs, or untruncated in the archive), they are 0.882, 0.812, and 0.626,

and full multi-state coverage is 0.530 (prefix values on the same episodes 0.828, 0.726, 0.582, and 0.505).

Table 12: First-pass protocol adherence and rule-based source-address diagnostics for NAVMEM with Qwen3.5-122B on BEAM 1M. Intervals denote 95% conversation-cluster bootstrap CIs (10,000 resamples). The 70 event-ordering episodes lack structured tool logs and are excluded from the first three rows.

Diagnostic	n	Rate	95% CI
$\geq$1 raw-read tool call	630	0.989	[0.979, 0.997]
Source-line citation	630	0.829	[0.802, 0.856]
Evidence-audit header	630	0.729	[0.687, 0.767]
$\geq$1 gold source cited	555	0.582	[0.546, 0.618]
All multi-state groups cited	204	0.515	[0.439, 0.591]

Table 13: Full BEAM 1M results across answering models and memory methods (Figure 3b ranks the Qwen3.5-122B block). We report macro scores over ten memory abilities: ABS=Abstention, CR=Contradiction Resolution, EO=Event Ordering, IE=Information Extraction, IF=Instruction Following, KU=Knowledge Update, MR=Multi-session Reasoning, PF=Preference Following, SUM=Summarization, and TR=Temporal Reasoning. Qwen results cover all 35 conversations. †Due to API cost constraints, the GPT-5.6 evaluation uses the first three conversations, fixed by index, for every method. Averages are computed from unrounded probe scores (the unweighted mean of rounded displayed values may differ by ± 0.001). The best score in each model block is shown in bold.

Model	Method	ABS	CR	EO	IE	IF	KU	MR	PF	SUM	TR	Avg.
Qwen3.6-35B-A3B	No Memory	**1.000**	0.000	0.251	0.000	0.068	0.000	0.013	0.024	0.000	0.021	0.138
	NativeRAG	0.714	0.652	0.450	0.562	0.527	0.593	0.548	0.654	0.427	0.439	0.557
	Mem0	0.661	0.757	0.496	0.660	0.697	0.611	0.592	**0.826**	0.498	0.579	0.638
	MemoryOS	0.646	0.529	0.510	0.558	0.738	0.461	**0.612**	0.803	**0.556**	0.262	0.567
	LIGHT	0.921	0.221	0.442	0.289	0.413	0.071	0.282	0.572	0.495	0.036	0.374
	A-MEM	0.679	0.511	0.381	0.638	0.634	0.611	0.528	0.735	0.453	0.350	0.552
	NAVMEM	0.771	**0.811**	**0.559**	**0.748**	**0.777**	**0.696**	0.573	0.723	0.507	**0.610**	**0.677**
Qwen3.5-122B	No Memory	**1.000**	0.000	0.251	0.000	0.071	0.000	0.013	0.055	0.000	0.025	0.141
	NativeRAG	0.707	0.637	0.437	0.600	0.586	0.614	**0.619**	0.745	0.464	0.418	0.583
	Mem0	0.650	0.738	0.458	0.600	0.613	0.732	0.563	0.807	0.443	0.542	0.615
	MemoryOS	0.643	0.605	0.453	0.565	0.683	0.557	0.524	0.768	0.446	0.288	0.553
	LIGHT	0.650	0.543	0.493	0.709	0.743	0.536	0.565	**0.829**	**0.572**	0.361	0.600
	A-MEM	0.654	0.527	0.416	0.626	0.677	0.625	0.503	0.801	0.485	0.339	0.565
	NAVMEM	0.789	**0.823**	**0.590**	**0.803**	**0.802**	**0.761**	**0.619**	0.760	0.517	**0.682**	**0.715**
GPT-5.6†	No Memory	**0.917**	0.000	0.258	0.000	0.500	0.000	0.382	0.875	0.000	0.000	0.293
	NativeRAG	0.667	0.312	0.450	0.806	0.569	0.833	0.764	0.736	0.271	**0.833**	0.624
	Mem0	0.667	0.312	0.525	0.514	0.708	**1.000**	0.750	0.736	0.319	0.500	0.603
	MemoryOS	0.667	0.417	0.500	0.556	0.833	0.833	0.826	0.861	0.395	0.542	0.643
	LIGHT	0.833	0.229	0.365	0.056	0.444	0.000	0.389	0.792	0.281	0.000	0.339
	A-MEM	0.583	0.188	0.285	0.694	0.583	0.833	**0.840**	0.861	0.264	0.625	0.576
	NAVMEM	0.833	**0.688**	**0.661**	**1.000**	**0.958**	**1.000**	0.806	**0.903**	**0.531**	**0.833**	**0.821**

Table 4 gives the model-scaling grid. Figure 4 and Table 14 report NAVMEM across all four BEAM context scales (100K to 10M tokens) and three answering models. Table 15 gives the complete LoCoMo token-F1 and judge accuracy results for every category and answering model, summarized in Table 5. Table 16 reports the detailed 10-ability breakdown of the BEAM 1M component ablation, and Table 17 its paired differences on both benchmarks.

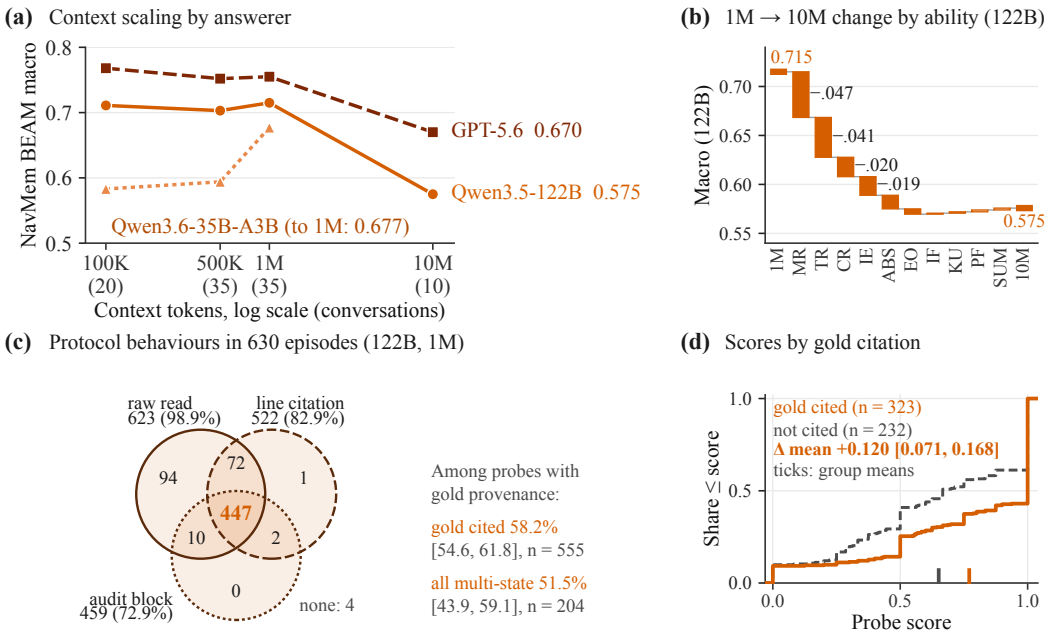


Figure 4: NAVMEM holds up to 1M tokens and drops at 10M, mostly on multi-session and temporal reasoning. **(a)** BEAM macro by context length (conversations in parentheses; Table 14); GPT-5.6 at 1M uses all 35 conversations; Qwen3.6-35B-A3B stops at 1M. **(b)** Qwen3.5-122B macro change from 1M to 10M by ability (unpaired). **(c)** Co-occurring protocol behaviors in 630 episodes; rates with 95% conversation-cluster intervals (Table 12). **(d)** Score ECDFs by gold-source citation; observational.

Table 14: NAVMEM on BEAM across context scales and answering models. We report macro scores over ten memory abilities: ABS=Abstention, CR=Contradiction Resolution, EO=Event Ordering, IE=Information Extraction, IF=Instruction Following, KU=Knowledge Update, MR=Multi-session Reasoning, PF=Preference Following, SUM=Summarization, and TR=Temporal Reasoning. Conversations per scale: 100K=20, 500K=35, 1M=35, 10M=10. Qwen3.5-122B covers every scale and Qwen3.6-35B-A3B covers 100K–1M (its 10M run answered most logged probes without memory tools and is not reported); at 1M, GPT-5.6 is reported for both the first three conversations and all 35. All context scales use an offline prebuilt memory ledger. Averages are computed from unrounded probe scores (the unweighted mean of rounded displayed values may differ by ± 0.001).

Scale	Model	ABS	CR	EO	IE	IF	KU	MR	PF	SUM	TR	Avg.
100K	Qwen3.6-35B-A3B	0.762	0.822	0.363	0.706	0.637	0.562	0.528	0.800	0.296	0.350	0.583
	Qwen3.5-122B	0.762	0.869	0.508	0.826	0.838	0.675	0.654	0.881	0.429	0.669	0.711
	GPT-5.6	0.775	0.906	0.636	0.831	0.950	0.662	0.676	0.931	0.540	0.769	0.768
500K	Qwen3.6-35B-A3B	0.564	0.827	0.370	0.787	0.574	0.718	0.507	0.775	0.433	0.387	0.594
	Qwen3.5-122B	0.607	0.852	0.502	0.866	0.682	0.814	0.630	0.899	0.590	0.592	0.703
	GPT-5.6	0.671	0.891	0.571	0.835	0.877	0.746	0.680	0.937	0.630	0.682	0.752
1M	Qwen3.6-35B-A3B	0.771	0.811	0.559	0.748	0.777	0.696	0.573	0.723	0.507	0.610	0.677
	Qwen3.5-122B	0.789	0.823	0.590	0.803	0.802	0.761	0.619	0.760	0.517	0.682	0.715
	GPT-5.6 (conv1–3)	0.833	0.688	0.661	1.000	0.958	1.000	0.806	0.903	0.531	0.833	0.821
	GPT-5.6 (conv1–35)	0.821	0.809	0.661	0.795	0.941	0.764	0.695	0.814	0.617	0.636	0.755
10M	Qwen3.6-35B-A3B	–	–	–	–	–	–	–	–	–	–	–
	Qwen3.5-122B	0.650	0.625	0.536	0.613	0.812	0.775	0.153	0.779	0.536	0.275	0.575
	GPT-5.6	0.750	0.738	0.669	0.725	0.838	0.775	0.222	0.771	0.678	0.537	0.670

Table 15: Full LoCoMo QA results (Table 5 condenses the ACC columns). Categories 1–4 use the official token-F1 scorer; for adversarial questions, a refusal judged correct receives F1$=1$ and any other answer F1$=0$. ACC is the GPT-5.5 judge score, and Overall pools all five categories. Qwen results cover all ten conversations (1,986 questions). †GPT-5.6 results cover the first three conversations (497 questions). LIGHT, Mem0, and A-MEM answer over memories built by Qwen3.6-35B-A3B in the GPT-5.6 block, and LIGHT and A-MEM do so in the Qwen3.5-122B block. MemoryOS was not evaluated on LoCoMo. The best score in each model block is shown in bold.

Model	Method	Multi-hop		Temporal		Open-domain		Single-hop		Adversarial		Overall	
		F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC
Qwen3.6-35B-A3B	No Memory	0.008	0.000	0.013	0.000	0.046	0.021	0.009	0.001	**0.998**	**0.998**	0.233	0.225
	NativeRAG	0.185	0.248	0.417	0.452	0.154	0.214	0.488	0.557	0.939	0.939	0.519	0.565
	Mem0	0.408	0.539	**0.570**	0.690	0.232	0.325	0.561	0.719	0.915	0.916	0.604	0.714
	LIGHT	**0.473**	**0.584**	0.494	0.578	0.268	0.378	0.620	0.744	0.839	0.839	**0.611**	0.698
	A-MEM	0.226	0.314	0.399	0.445	0.203	0.239	0.402	0.494	0.922	0.925	0.483	0.545
	NAVMEM	0.432	0.578	0.264	**0.704**	**0.385**	**0.525**	**0.682**	**0.807**	0.832	0.833	0.598	**0.750**
Qwen3.5-122B	No Memory	0.008	0.000	0.013	0.000	0.039	0.000	0.009	0.000	**1.000**	**1.000**	0.233	0.225
	NativeRAG	0.169	0.245	0.439	0.500	0.128	0.169	0.487	0.566	0.955	0.957	0.522	0.579
	Mem0	**0.437**	**0.574**	**0.468**	0.550	0.182	0.260	0.525	0.693	0.937	0.937	0.579	0.687
	LIGHT	0.373	0.557	0.444	0.545	0.148	0.227	0.545	0.714	0.926	0.929	0.571	0.689
	A-MEM	0.205	0.276	0.383	0.466	0.117	0.157	0.370	0.482	0.924	0.925	0.461	0.534
	NAVMEM	0.425	0.557	0.352	**0.646**	**0.336**	**0.486**	**0.668**	**0.804**	0.926	0.927	**0.624**	**0.755**
GPT-5.6†	No Memory	0.001	0.000	0.003	0.000	0.073	0.024	0.009	0.000	**1.000**	**1.000**	0.233	0.226
	NativeRAG	0.207	0.280	0.574	0.694	0.152	0.171	0.416	0.581	0.821	0.826	0.494	0.595
	Mem0	0.416	**0.653**	0.658	**0.833**	0.199	0.353	0.499	0.780	0.768	0.777	0.564	0.752
	LIGHT	0.442	0.624	0.655	0.761	0.200	0.341	0.529	0.802	0.652	0.670	0.553	0.719
	A-MEM	0.217	0.329	0.522	0.650	0.155	0.282	0.371	0.551	0.804	0.804	0.464	0.581
	NAVMEM	**0.485**	0.592	**0.667**	0.772	**0.328**	**0.682**	**0.627**	**0.832**	0.911	0.911	**0.664**	**0.797**

Table 16: Detailed 10-ability component ablation on BEAM 1M (Qwen3.5-122B, 35 conversations, 700 questions). ABS=Abstention, CR=Contradiction Resolution, EO=Event Ordering, IE=Information Extraction, IF=Instruction Following, KU=Knowledge Update, MR=Multi-session Reasoning, PF=Preference Following, SUM=Summarization, and TR=Temporal Reasoning. Best score per column is bolded. Averages are unweighted macro scores across the 10 abilities. The two evidence pairs are omitted (Appendix G).

Configuration	ABS	CR	EO	IE	IF	KU	MR	PF	SUM	TR	Macro
No Memory	**1.000**	0.000	0.251	0.000	0.071	0.000	0.013	0.055	0.000	0.025	0.141
Evidence only	0.839	0.609	0.565	0.722	0.621	0.532	0.550	0.719	0.543	0.587	0.629
Continuity only	0.904	0.352	**0.646**	0.419	0.797	0.243	0.546	**0.787**	**0.592**	0.355	0.564
Dynamics only	0.857	0.520	0.461	0.387	0.516	0.500	0.366	0.692	0.245	0.256	0.480
Continuity + Dynamics	0.911	0.648	0.518	0.539	0.724	0.671	0.497	0.629	0.391	0.327	0.586
Full NAVMEM	0.789	**0.823**	0.590	**0.803**	**0.802**	**0.761**	**0.619**	0.760	0.517	**0.682**	**0.715**

Table 17: Paired differences from the full NAVMEM system in the component ablation (Qwen3.5-122B; arm minus full system; 95% conversation-cluster bootstrap CIs, 10,000 resamples, seed 20260925). n: questions compared. †Excludes the questions this arm answered in one call without the transcript (272 for Evidence + Continuity; 377 for Evidence + Dynamics, including every question of conversations 22–35, so its interval uses 21 conversations).

	BEAM 1M			LoCoMo		
Configuration	n	Δ	95% CI	n	Δ	95% CI
No Memory	700	-0.573	$[-0.603, -0.543]$	1,986	-0.531	$[-0.554, -0.509]$
Evidence only	700	-0.086	$[-0.110, -0.062]$	1,986	-0.090	$[-0.113, -0.067]$
Continuity only	700	-0.151	$[-0.178, -0.124]$	1,986	-0.401	$[-0.440, -0.367]$
Dynamics only	700	-0.235	$[-0.260, -0.209]$	1,986	-0.235	$[-0.263, -0.209]$
Evidence + Continuity†	428	-0.046	$[-0.076, -0.017]$	1,986	-0.056	$[-0.069, -0.042]$
Evidence + Dynamics†	323	-0.043	$[-0.078, -0.007]$	1,986	-0.004	$[-0.014, 0.006]$
Continuity + Dynamics	700	-0.129	$[-0.149, -0.107]$	1,986	-0.189	$[-0.218, -0.165]$

H LIMITATIONS

Read-side verification is prompt-instructed, because the harness retries format, no-tool, and route-quality failures but never checks that a cited span supports the claim. All runs use an offline prebuilt ledger, so we did not benchmark online capture, crash recovery, or concurrent updates. The archived records do not show whether the optional embedding fusion was active, and no ablation isolates it. The continuity-only and dynamics-only arms use no search tool and no embedding, and in two pair arms the tool loop failed on part of the BEAM questions (Appendix G). Only the BEAM comparison with Mem0 has a second judge, and GPT-5.6 baseline comparisons use three conversations and support comparisons within that prefix only. Scores drop at 10M tokens, so the 1M result does not carry over to arbitrary context lengths.