LESS CONTEXT, MORE ACCURACY: STRUCTURED LONG-TERM MEMORY BEYOND EQUAL-BUDGET RAG

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term memory for LLM agents is often justified against replaying the entire interaction history, which is expensive and can become less accurate as distractors accumulate. Yet outperforming full context does not establish that *structured* memory adds value beyond ordinary retrieval: a token-matched RAG baseline may recover the same gain. Existing evaluations also change the reader, judge, or context budget together with the memory system, making gains difficult to attribute and reproduce. We present ENGRAM, an open-source, dual-process memory engine built on a bi-temporal data model. A fast write path appends lossless episodes without an LLM on the critical path; an asynchronous consolidation path extracts atomic $(\textit{subject}, \textit{predicate}, \textit{object})$ facts, builds a bi-temporal knowledge graph, and resolves contradictions *without* an LLM call per fact—invalidating, never deleting, so every fact retains provenance and a supersession chain. A hybrid read path fuses dense, lexical, graph, and recency/salience signals, applies a point-in-time ("as-of") temporal filter, and assembles a compact, provenance-tagged context. We evaluate frozen configurations on all 500 questions of LONGMEMEVAL$_S$ with the official category-specific judge prompts. Streamlined ENGRAM reaches **78.8%**, significantly outperforming an equal-budget dense+BM25 RAG control at **72.4%** (+6.4 points; McNemar $p = 0.001$) and full context at **61.6%** (+17.2 points), while using 6.7k rather than 103.5k mean context tokens (15.5$\times$ compression). The 86.0% annotated-evidence oracle is +7.2 points higher. The predefined 440-question held-out partition preserves the margins. Five single-variable interventions show that gains are category-specific and that an evidence planner and hierarchy blocks reduce aggregate accuracy; repeated generations and a 2×2 reader–judge cross test robustness. A separate streaming experiment on all 500 LONGMEMEVAL$_M$ questions measures the shared episode index without conflating retrieval recall with QA accuracy. Every reported value is generated from fingerprinted, append-only context, answer, and judgment artifacts.

1 INTRODUCTION

LLM agents are stateless across sessions. The pragmatic fix—concatenate the whole conversation history into the prompt—scales badly on three axes at once: token cost grows linearly with history, latency follows, and accuracy *degrades* as irrelevant turns crowd the window and the model is forced to locate a needle among distractors (Liu et al., 2024). A long-term memory layer that stores, structures, and selectively retrieves the past is the natural alternative, and a growing body of systems pursues it (Packer et al., 2023; Park et al., 2023; Chhikara et al., 2025; Rasmussen et al., 2025; Gutiérrez et al., 2024); see Du (2026) for a recent (2026) survey.

Two gaps motivate this work. First, **attribution**: beating full context can show the value of context selection, but not whether structured memory contributes beyond a strong retriever under the same token budget. The decisive control is therefore equal-budget RAG, with the reader, judge, prompt, and questions held fixed. Second, **reproducibility**: memory benchmarks use different ingestion, answer prompts, and judges, so a single system can appear tens of points apart depending on the source, and different papers give contradictory orderings. In a field where every number is contested, an audit-oriented harness anyone can re-run is itself a contribution.

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.

We ask a focused question: *after generic context selection is controlled for, does structured long-term memory still improve answer quality?* We answer it with ENGRAM, an open-source memory engine, and an in-repo evaluation harness. The system composes typed memory, a bi-temporal knowledge graph, multi-signal retrieval fusion, salience decay, and asynchronous consolidation; the harness isolates this structured path from both generic retrieval and full-history replay.

Contributions.

1. **A controlled test beyond equal-budget RAG** (§5) on all 500 LONGMEMEVAL$_S$ questions. Streamlined ENGRAM achieves margins of +6.4 points over dense+BM25 RAG ($p = 0.001$) and +17.2 points over full context, while using 15.5× fewer prompt tokens than full context; the annotated-evidence oracle is only +7.2 points higher.
2. **A hybrid, bi-temporal memory design** (§3) that combines persistent facts, raw source chunks, graph links, provenance, and reconciled temporal state. Contradictions are invalidated rather than overwritten, preserving supersession chains and provenance without an LLM call in the common case.
3. **Mechanistic evidence about what structured memory needs** (§5). Five frozen interventions reveal category-specific effects and reject two plausible additions: the evidence planner and hierarchy blocks reduce aggregate accuracy. Three generations and a 2×2 reader–judge cross separately test robustness.
4. **An audit-oriented, reproducible harness** (§4) with official judging prompts, budget-matched contexts, a predefined held-out split, immutable context/answer/judgment artifacts, paired tests, raw logs, and a separate full LONGMEMEVAL$_M$ retrieval-scaling experiment.

2 RELATED WORK

Memory systems for LLM agents. MemGPT (Packer et al., 2023) frames the LLM as an operating system that pages memory in and out of a limited context window. Generative Agents (Park et al., 2023) introduce a memory stream with importance, recency, and relevance scoring plus periodic reflection. Mem0 (Chhikara et al., 2025) targets production agents with scalable extract-and-store memory. Zep/Graphiti (Rasmussen et al., 2025) is the closest in spirit to ENGRAM: a temporally-aware knowledge-graph memory with bi-temporal edges. HippoRAG (Gutiérrez et al., 2024) uses a graph plus personalized PageRank for multi-hop retrieval, and its successor HippoRAG 2 (Gutiérrez et al., 2025) reframes the same machinery as non-parametric continual memory. The most recent agentic-memory systems add LLM-driven control over how memories are written and organized: A-MEM (Xu et al., 2025) builds an interlinked, Zettelkasten-style note network; MemoryOS (Kang et al., 2025) imposes an operating-system-style short/mid/long-term hierarchy; and GAM (Wu et al., 2026) decouples memory encoding from consolidation over hierarchical graphs. ENGRAM differs in *composing* a bi-temporal graph with a hybrid (facts + raw chunks) read path and an explicit, cheap conflict-resolution policy. More importantly for attribution, we compare this structured path against an equal-budget dense+BM25 RAG control, rather than crediting all gains over full history to the memory design.

Long context and the cost of distractors. "Lost in the middle" (Liu et al., 2024) shows that LLMs use long contexts unevenly and that accuracy degrades when relevant evidence is buried among distractors. This is the mechanism our headline result exploits: a filtered, precisely retrieved slice can be *more* accurate than the full window, not merely cheaper.

Benchmarks. LONGMEMEVAL (Wu et al., 2025) evaluates chat assistants on long-term interactive memory across categories (single/multi-session, knowledge-update, temporal reasoning, preference, and abstention), with category-specific judge prompts. More recent benchmarks extend the setting to multimodal histories (Mem-Gallery (Bei et al., 2026)). We use the complete LONGMEMEVAL$_S$ release for end-to-end QA and the longer LONGMEMEVAL$_M$ release for retrieval-scaling measurements.

Retrieval. ENGRAM's read path builds on dense retrieval (Karpukhin et al., 2020) with modern dense sentence embeddings (the BGE family (Xiao et al., 2024); we use the English

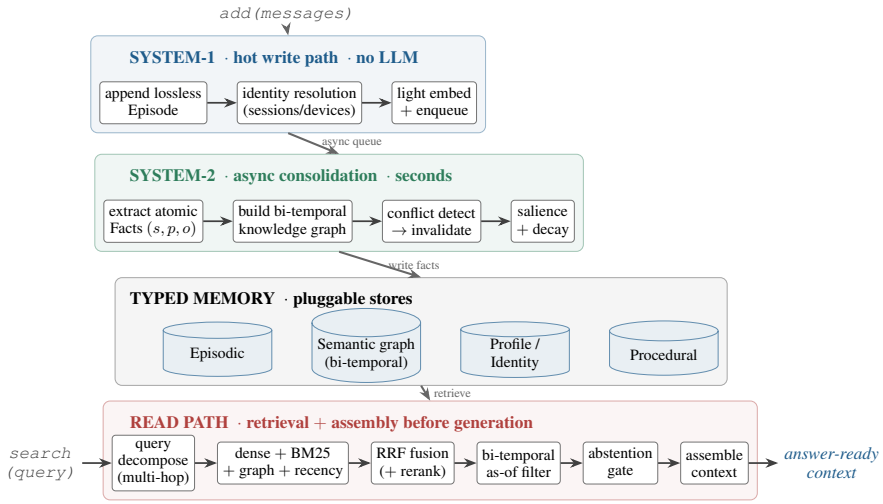

Figure 1: The ENGRAM dual-process architecture: a hot write path (SYSTEM-1) that never blocks on an LLM; an asynchronous consolidation path (SYSTEM-2) that extracts facts, builds the bi-temporal graph and resolves conflicts non-destructively; a typed memory over pluggable stores; and a hybrid read path (dense + lexical + graph + recency, fused by RRF) with an as-of filter and an abstention gate.

`bge-small-en-v1.5`), classical lexical scoring (BM25 (Robertson and Zaragoza, 2009)), and Reciprocal Rank Fusion (Cormack et al., 2009), in the retrieval-augmented generation tradition (Lewis et al., 2020). The dual-process framing draws on the System-1/System-2 distinction in cognitive science (Kahneman, 2011), and salience decay on the classical forgetting curve (Ebbinghaus, 1913).

3 THE ENGRAM SYSTEM

ENGRAM is a dual-process memory system: a fast online write path (System-1) and a slow asynchronous consolidation path (System-2) that writes into a typed, bi-temporal memory, which a hybrid read path queries (Figure 1).

3.1 BI-TEMPORAL DATA MODEL

Two time axes are first-class everywhere, which is what makes knowledge-updates and "as-of" queries intrinsic rather than bolted-on.

- **Episode** — a raw, lossless turn/event, stamped with *event time* (when it happened in the world) and *ingested-at* (transaction time, when we recorded it).
- **Fact** — an atomic $(subject, predicate, object)$ claim with surface text, an embedding, a *salience* and *confidence*, and *provenance* (the source episode ids). It carries two time axes: *valid time* (`valid_at` / `invalid_at`, when the claim is true in the world) and *transaction time* (`created_at` / `expired_at`, when we learned or retracted it), plus a `supersedes` pointer to the fact it replaces.
- **Entity / Relation** — graph nodes and edges; edges carry the same bi-temporal stamps.

The invariant: *never hard-delete a contradicted fact—invalidate it* (set `invalid_at`) and keep the history. Every fact can therefore answer "where did this come from?" and "what did it replace?" (Figure 2).

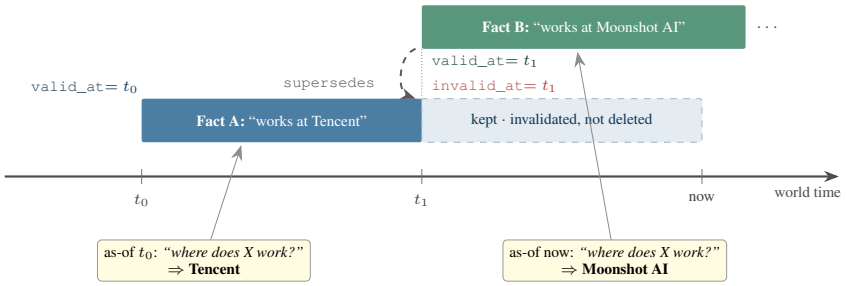

Figure 2: Bi-temporal facts make contradictions and "as-of" queries first-class: when Fact B arrives at t_1, ENGRAM sets A.`invalid_at` $= t_1$ and B.`supersedes` = A—A is *invalidated, not deleted*, so history and provenance survive—and a point-in-time query resolves against the fact valid at that time, which is what the knowledge-update and temporal categories require.

3.2 SYSTEM-1: THE HOT WRITE PATH

On `add(messages)`, ENGRAM appends a lossless episode, resolves identity (linking a user/entity across sessions and devices), computes a light embedding, and enqueues the episode for consolidation. No LLM runs on this path, so writes do not wait for model generation.

3.3 SYSTEM-2: ASYNCHRONOUS CONSOLIDATION

Off the critical path, the consolidation engine (i) extracts atomic facts from episodes (rule-based by default, LLM-based when a model is configured), (ii) builds the bi-temporal knowledge graph of entities and relations, (iii) detects conflicts and invalidates superseded facts, and (iv) scores salience and applies decay/reinforcement so unreinforced memories fade and the store stays small and fast. Hierarchical abstraction (session summaries → profile) runs here as well.

3.4 CHEAP-THEN-ESCALATE CONFLICT RESOLUTION

When a new fact arrives for an existing (*subject*, *predicate*) slot with a different object, ENGRAM resolves it in increasing order of cost. First, cheap signals: an exact **slot match** signals a likely update, **embedding similarity** catches the same attribute under a different free-form predicate, and **content subsumption** (one claim $\subseteq$ the other) separates contradiction from elaboration. Second, if the claims are clearly contradictory and temporally ordered, the old one is invalidated (`old.invalid_at` $\leftarrow$ `new.valid_at`) and `new.supersedes` $\leftarrow$ `old.id`—with *no LLM call*. Only genuinely ambiguous cases escalate to an LLM adjudicator. This is the cost win over systems that invoke an LLM on every fact, while preserving temporal correctness and a complete audit trail.

3.5 THE HYBRID READ PATH

On `search(query)`, ENGRAM (1) understands the query and, for multi-hop questions, decomposes it into sub-queries; (2) retrieves in parallel through four complementary channels—dense semantic, BM25 lexical, graph n-hop from the query's entities, and recency/salience; (3) fuses the ranked lists with Reciprocal Rank Fusion (Cormack et al., 2009) and an optional cross-encoder rerank over the top-k (off by default); (4) applies a bi-temporal "as-of" filter (what we believed was true at time T); (5) passes an abstention gate that declines to answer when the evidence is absent; and (6) assembles a deduplicated, provenance-tagged, token-budgeted context. The per-item score combines all signals:

$$\begin{aligned}\text{score}(\textit{item} \mid q) = w_{\text{sem}}\cos(q, \textit{item}) + w_{\text{lex}}\,\text{bm25}(q, \textit{item}) + w_{\text{graph}}\,\text{prox}(\textit{item}) \\ + w_{\text{rec}}\,e^{-\Delta t/\tau} + w_{\text{sal}}\,\text{sal}(\textit{item}),\end{aligned} \tag{1}$$

where the weights are configuration-driven rather than hand-set per question. Crucially, the assembled context is *hybrid*: it contains both conflict-resolved bi-temporal facts and the most relevant raw session chunks. This composition preserves verbatim detail that fact extraction can omit; the full-set interventions isolate selected read-path mechanisms under the same token budget.

Every external dependency—LLM, embedder, vector, graph and lexical store—sits behind an interface with a zero-dependency offline fallback, so the loop runs with no API keys.

4 EXPERIMENTAL SETUP

Benchmark. We use all 500 questions of LONGMEMEVAL$_S$ (Wu et al., 2025), spanning multi-session, temporal-reasoning, knowledge-update, single-session user/assistant, and preference tasks. The complete history averages 103.5k tokens in our tokenizer. We report both the official 500-question aggregate and a 440-question held-out partition. Before any final run, we sampled a fixed 60-question development set stratified to 10 items per category (seed 20260913); every retrieval, model, and context-assembly candidate was frozen on those 60 items before final grading. We report the remaining 440 question IDs separately to show that full-set conclusions are not driven by the development subset. Accuracy uses the benchmark's official category-specific prompts, including its temporal, knowledge-update, preference, and unanswerable rules. The full-context safety cap is 600,000 characters; the largest materialized history is 521,012 characters, so none of the 500 full histories is truncated.

Models. The local embedder is `BAAI/bge-small-en-v1.5`; reranking is disabled. The frozen System-2 extractor is `deepseek-v4-flash-ga-260731`. Reader 1 is `doubao-seed-2-1-pro-260628` and Reader 2 is `deepseek-v4-flash-ga-260731`; Judge 1 and Judge 2 use the same two model versions in reverse order. Read-time LLM query expansion and aggregation are disabled, leaving only deterministic planning on the read path. Every artifact records the requested endpoint, actually served model, token usage, and latency.

Systems. Four systems establish the controlled comparison: (i) `oracle_evidence`, containing only the benchmark-annotated evidence sessions under the same reader prompt; (ii) `full_context`, containing the complete history; (iii) `rag_token_matched`, a strong dense+BM25 RRF baseline with the same BGE embedder and 8k-token cap as ENGRAM but no facts, graph, or temporal state; and (iv) `engram_final`, the frozen hybrid path with conflict-resolved facts, direct source chunks, provenance-guided source expansion, deterministic evidence planning, graph retrieval, and bi-temporal updates. The final path deliberately omits session-summary and profile blocks, which the development criterion rejected. The oracle is a reader ceiling conditioned on human evidence labels, not a deployable retrieval system: unlike every automatic method, it is told which sessions contain the answer.

Five full-set interventions inherit `engram_final` and change one mechanism: remove provenance-guided source expansion, remove the evidence planner, remove graph retrieval, remove write-time temporal reconciliation, or add the hierarchy blocks back. We call the planner-free intervention `engram_final_no_planner`; it is the strongest frozen automatic configuration in the final matrix. The first intervention keeps the fixed direct-detail chunks, so it isolates provenance-guided expansion rather than collapsing the system into facts-only QA. The bi-temporal intervention disables write-time supersession and invalidation while retaining dated raw chunks and the common date-aware reader prompt; it therefore isolates reconciled memory state rather than removing time information from the question-answering task.

Retrieval configuration. The hybrid read path fuses five ranked signals by weighted Reciprocal Rank Fusion ($k_{\text{RRF}} = 60$) with the Eq. (1) weights $w_{\text{sem}} = 1.0$, $w_{\text{lex}} = 0.6$, $w_{\text{graph}} = 0.8$, $w_{\text{rec}} = 0.3$, $w_{\text{sal}} = 0.25$. The context builder extracts facts from at most 8 retrieved sessions, retrieves up to 15 facts, includes 2 direct session chunks, and enforces an 8,000-token cap with the `o200k_base` tokenizer. On dev60, BGE-small attained 98.33% macro evidence-session Recall@15, 100% multi-session Recall-all, and 100% latest-evidence hit rate for knowledge updates. A BGE reranker reduced these to 91.61%, 90%, and 90%, respectively, so the frozen configuration uses no reranker.

Randomness and cross-model controls. Contexts are materialized once and reused. Reader 1 generates `engram_final` and `full_context` three times, and Reader 2 evaluates `engram_final`, 8k-budget RAG, and full context once. Both judges independently grade every saved answer, including `engram_final_no_planner`, while the repeated-generation and 2×2 reader–judge checks target the complete configuration without rebuilding contexts.

Artifacts and statistics. The frozen manifest fingerprints the dataset revision and SHA-256, runtime code tree, tokenizer, prompts, models, dependencies, split, and every hyperparameter. Context, answer, and judgment are separate append-only JSONL artifacts joined by SHA-256; retries fill only missing keys. Publication requires exactly 4,500 contexts, 8,000 answers, and 16,000 judgments with no missing or unexpected key. McNemar's exact test and a 20,000-sample paired bootstrap quantify comparisons. All paper values and figures are generated directly from these artifacts by `paper/final_results.py`.

LongMemEval-M. We separately stream all 500 questions of the 2.745GB LONGMEMEVAL$_{\text{M}}$ file and measure the shared raw-episode dense+BM25 RRF layer over 502 sessions per question. This retrieval-only experiment makes no extractor, reader, or judge calls and supports only episode-index scaling claims; its Recall@k is not QA accuracy.

Evaluation controls. Appendix B specifies the controls applied to every reported comparison: isolated and budget-matched contexts, frozen random boundaries, official grading semantics, paired systems, artifact fingerprints, immutable keys, and explicit failure accounting.

5 RESULTS

Headline. Table 1 and Figure 3 report the strongest frozen automatic configuration on all 500 questions. Streamlined ENGRAM reaches **78.8%**, compared with 72.4% for 8k-budget RAG and 61.6% for full context. Its paired margins are +6.4 and +17.2 points, respectively; the exact McNemar p-values are 0.001 and 8.8×10^{-13}. Paired bootstrap 95% intervals are [+2.8, +10.2] and [+12.6, +21.8] points. On the predefined 440-question partition, ENGRAM scores 78.6% versus 72.5% and 60.9%, with margins of +6.1 and +17.7 points. Its agreement with the full-set result shows that the aggregate is not driven by the 60 development questions. The annotated-evidence oracle reaches 86.0%, +7.2 points above ENGRAM, but is told the answer-bearing sessions. Pairwise, 364 questions are correct for both, 66 only for the oracle, 30 only for ENGRAM, and 40 for neither; this separates the remaining evidence-selection headroom from the reader ceiling.

Table 1: LONGMEMEVAL$_{\text{S}}$, all 500 questions. Accuracy uses the primary reader, primary judge, and first frozen generation; brackets give Wilson 95% intervals. Context tokens are measured before the common answer prompt.

System	Accuracy (95% CI)	Held-out accuracy	Mean context tokens
annotated-evidence oracle	86.0% [82.7, 88.8]	85.5%	5.1k
ENGRAM (`engram_final_no_planner`)	**78.8%** [75.0, 82.2]	78.6%	**6.7k**
ENGRAM (`engram_final`)	75.4%	74.5%	7.1k
8k-budget hybrid RAG	72.4% [68.3, 76.1]	72.5%	8.0k
full context	61.6% [57.3, 65.8]	60.9%	103.5k

Per-category. Table 2 and Figure 4 break the primary comparison down by the benchmark's six question categories. Reporting RAG beside full context distinguishes gains from generic context filtering from gains attributable to ENGRAM's structured memory path. These category values are descriptive. Relative to equal-budget RAG, the largest shifts are +36.7 points on preference questions and +9.0 points on temporal reasoning, versus -3.6 points on single-session assistant questions. Exploratory exact paired tests sharpen the pattern: preference questions show 11 Engram-only wins and 0 RAG-only wins ($p = 9.8 \times 10^{-4}$); the temporal comparison has $p = 0.023$. Against full context, the gains on multi-session and temporal questions are +27.1 and +24.1 points ($p = 1.2 \times 10^{-6}$ and $p = 3.3 \times 10^{-6}$, respectively).

Single-variable interventions. Table 3 removes one frozen read-path mechanism at a time; positive drops mean that the complete system is more accurate. "Without provenance expansion" disables provenance-evidence injection and provenance-guided chunk promotion while retaining fixed direct-detail chunks, so it is not a facts-only system. The hierarchy row restores the session-summary blocks rejected on the development split. The planner-free intervention is the strongest automatic configuration at 78.8%, exceeding the complete configuration's 75.4% by 3.4 points. The remaining effects are

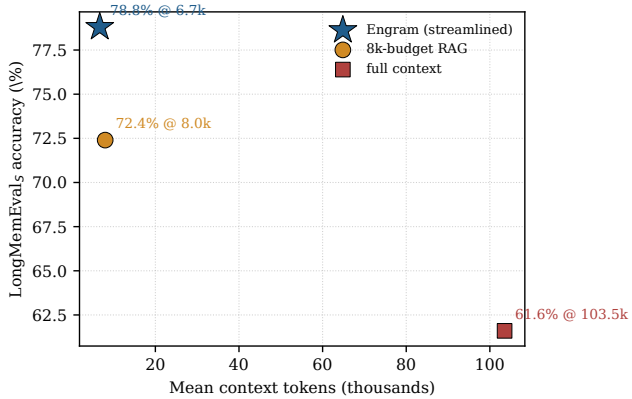

Figure 3: Accuracy versus mean context tokens for the primary 500-question comparison. All systems use the same reader, answer prompt, and judge; equal-budget RAG isolates the gain from structured memory beyond generic context selection.

Table 2: Per-category accuracy (%) on all 500 questions. Unanswerable items remain in their official host categories.

Category	ENGRAM	8k RAG	Full context	n
knowledge update	88.5	88.5	82.1	78
multi-session	65.4	58.6	38.3	133
single-session assistant	94.6	98.2	94.6	56
single-session preference	70.0	33.3	46.7	30
single-session user	88.6	85.7	80.0	70
temporal reasoning	76.7	67.7	52.6	133

category-specific rather than uniformly additive: provenance expansion contributes +6.7/+4.5/+2.6 points on preference/temporal/knowledge-update questions, and graph retrieval contributes +2.3 points on temporal reasoning, while other removals can improve aggregate accuracy. The hierarchy decision is unambiguous: the frozen complete system exceeds the hierarchy intervention by 18.8 points.

Table 3: Primary-run interventions on all 500 questions. "Wins/losses" counts paired questions on which only ENGRAM or only the intervention is correct.

Intervention	Accuracy (%)	ENGRAM minus intervention	Wins / losses
without provenance expansion	75.8	-0.4	24 / 26
without evidence planner	78.8	-3.4	27 / 44
without graph retrieval	77.0	-1.6	14 / 22
without temporal reconciliation	78.0	-2.6	10 / 23
with hierarchy blocks	56.6	+18.8	122 / 28

Decoding and evaluator robustness. Table 4 reports three independent answer generations with the primary reader and judge. Across them, `engram_final` averages 75.7% (range 75.4%–76.0%) and full context averages 61.0% (range 60.6%–61.6%); every paired margin is positive, ranging from +13.8 to +15.2 points. Table 5 crosses two dated reader endpoints with two dated judge endpoints for `engram_final` on the frozen contexts. Judge agreement over the complete judgment matrix is 96.8% (Cohen's $\kappa = 0.924$). These controls are reported separately rather than pooled into the primary score.

LongMemEval-M retrieval scaling. Across 500 questions and 250948 streamed sessions, the shared raw-episode dense+BM25 RRF layer obtains Recall@5/10/15 of 74.5%/84.6%/89.2%; all

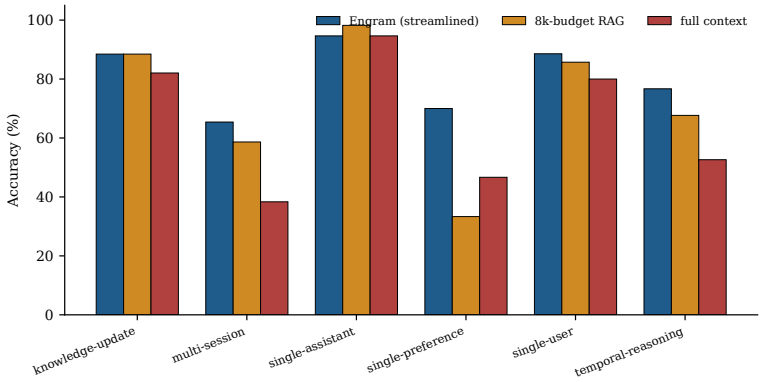

Figure 4: Per-category accuracy for ENGRAM, 8k-budget RAG, and full context on the same questions.

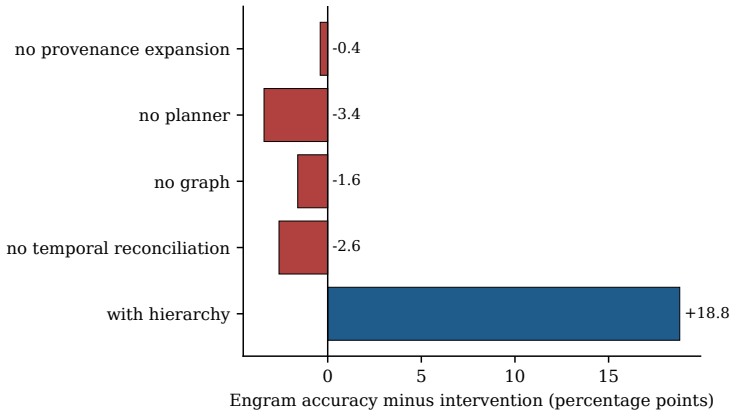

Figure 5: Accuracy of the complete frozen system and each single-variable intervention.

annotated evidence is present in the top 15 for 83.4% of questions. Per-question index construction is 27.94 s p50 and 87.81 s p95; query latency is 1138.5 ms p50 and 2966.4 ms p95. The logical float32 index averages 0.74 MiB and returns 40360 tokens. This isolates the retrieval layer at the M scale; it is not an end-to-end QA score for the consolidated graph.

Context and answer-time efficiency. ENGRAM supplies 6.7k mean context tokens, compared with 8.0k for the 8k-budget control and 103.5k for full context: a 15.5× reduction from the latter. With contexts already materialized, primary-reader answer latency is 3.5 s/7.8 s p50/p95 for ENGRAM and 24.3 s/54.8 s for full context. Build and answer latency are reported separately because the former includes asynchronous consolidation and the latter is dominated by the remote reader.

6 DISCUSSION

Full context contains every answer-bearing session, yet loses +17.2 points to ENGRAM under the same reader and judge. The largest margins—+27.1 points on multi-session questions and +24.1 on temporal reasoning—occur exactly where a reader must connect sparse evidence across a long, distractor-heavy history. This is an accuracy result, not only a compression result: selecting evidence changes what the fixed reader can recover. The annotated-evidence oracle then cleanly separates the remaining retrieval headroom from the reader's own ceiling.

The 8k-budget RAG control recovers +10.8 points over full context, showing that context selection itself accounts for much of the gain. Streamlined ENGRAM adds a statistically significant +6.4 points

Table 4: Primary-reader answer-generation repeats for the complete configuration on all 500 questions.

Generation	`engram_final` (%)	Full context (%)	Difference
1	75.4	61.6	+13.8
2	76.0	60.8	+15.2
3	75.8	60.6	+15.2

Table 5: Reader–judge cross-check on frozen generation 1. Model identifiers appear in §4; entries are accuracy (%).

Pair	`engram_final`	8k RAG	Full context
Reader 1 / Judge 1	75.4	72.4	61.6
Reader 1 / Judge 2	72.2	70.0	58.6
Reader 2 / Judge 1	72.8	71.6	56.4
Reader 2 / Judge 2	69.6	68.6	55.4

overall ($p = 0.001$), with its clearest advantage on preference questions and a further temporal gain. This concentration matches the intended role of persistent preference facts, provenance, graph links, and reconciled state rather than generic semantic search. The interventions reject two tempting design choices: the deterministic evidence planner reduces aggregate accuracy by 3.4 points, and hierarchy blocks reduce it by 18.8 points. The streamlined configuration therefore keeps facts, graph links, temporal state, provenance expansion, and raw source chunks but omits both additions. The remaining intervention map is category-selective rather than additive, so we report paired and category-specific effects. LongMemEval-M separately measures the episode retriever at the longer-history scale and is not presented as end-to-end QA.

7 Conclusion

Under a controlled comparison with the same reader, judge, prompt, and context budget, structured memory adds +6.4 accuracy points over dense+BM25 RAG on all 500 LongMemEval$_S$ questions ($p = 0.001$). Streamlined Engram reaches 78.8% versus 72.4% for RAG and 61.6% for full context, while using 15.5× fewer context tokens than full context; the annotated-evidence oracle is +7.2 points higher. The predefined held-out split and single-variable interventions support this attribution; repeated generations and the reader–judge cross independently confirm the direction for the complete configuration. The separate LongMemEval$_M$ experiment extends the evidence to episode-index retrieval at scale. The system, harness, and logs are included in the anonymous supplementary package.

AI Use Statement

Generative AI tools assisted implementation and refactoring, evaluation-harness development, experiment design feedback, statistical and figure code, interpretation of results, and drafting and editing this manuscript. Separately, the answerer, fact extractor, and judge are measured LLM components named with model identifiers in §4. The author reviewed the generated code and text, checked the reported statistics against the committed logs (`paper/check_numbers.py`), and takes full responsibility for the experiments, claims, text, and code.

Reproducibility Statement

Every measured result is backed by the frozen manifest and append-only logs under `results/iclr2027/final_v1/`. Each of the 4,500 context records contains its exact text, tokenizer count, build telemetry, and SHA-256; each of the 8,000 answer records points to that context hash; and each of the 16,000 judgments points to the saved prediction hash. The artifact checker rejects missing, unexpected, duplicate-success, wrong-config, or broken-link keys before

emitting any paper number. The end-to-end library and unit tests retain deterministic no-key fallbacks; reproducing the reported benchmark matrix requires access to the named model versions in §4.

The complete S experiment, statistics, and figures are reproduced by:

```
./scripts/run_iclr_final.sh \
  && python paper/final_results.py \
  && python paper/figs/make_final_figs.py
```

The retrieval-only M experiment is reproduced by:

```
python scripts/run_longmemeval_m_retrieval.py --data m \
  --embedder bge-small \
  --out results/iclr2027/longmemeval_m/episode_retrieval.json
```

Both result builders also generate the TeX macros consumed by this manuscript; measured values are not transcribed by hand. The appendix reproduces the answer and judge prompts. Code, manifests, raw logs, and the anonymized supplement are provided with the submission; the public repository URL is included only in the non-anonymous build.

ETHICS STATEMENT

A long-term memory layer necessarily persists personal data across sessions, which raises real privacy obligations. Two of ENGRAM's core design choices are mitigations as much as features: *non-destructive invalidation with full provenance* yields an auditable trail of what was believed, when, and from which source, and *salience decay* reduces the retrieval prominence of unreinforced details. Non-destructive invalidation is a statement about *contradiction*, not about *erasure*: a contradicted fact is kept, invalidated, so that the history stays auditable, but the store also exposes explicit deletion—a single fact, or everything held about a user—so that a "right to be forgotten" request is a supported operation rather than a best-effort one. Because every component has a zero-dependency offline fallback, ENGRAM can run fully locally, so no user data need leave the operator's machine; the AGPL-3.0 release also keeps the implementation inspectable and self-hostable. The principal risks are the obverse of the benefits: a memory store concentrates sensitive information and must be secured, access-controlled, and scoped to genuine consent; and a memory that confidently surfaces a *stale* or *wrong* fact can mislead, which is precisely why the bi-temporal model and the abstention gate (decline when memory lacks the answer) are first-class rather than optional. All experiments use the public LONGMEMEVAL release, whose conversations are synthetic; no private user data were collected or used.

REFERENCES

Yuanchen Bei, Tianxin Wei, Xuying Ning, Yanjun Zhao, Zhining Liu, Xiao Lin, Yada Zhu, Hendrik Hamann, Jingrui He, and Hanghang Tong. Mem-Gallery: Benchmarking multimodal long-term conversational memory for MLLM agents. *arXiv preprint arXiv:2601.03515*, 2026.

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

Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 758–759, 2009.

Pengfei Du. Memory for autonomous LLM agents: Mechanisms, evaluation, and emerging frontiers. *arXiv preprint arXiv:2603.07670*, 2026.

Hermann Ebbinghaus. *Memory: A Contribution to Experimental Psychology*. Teachers College, Columbia University, 1913. Original work published 1885.

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 (NeurIPS)*, 2024.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From RAG to memory: Non-parametric continual learning for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 21497–21515. PMLR, 2025. URL https://proceedings.mlr.press/v267/gutierrez25a.html.

Daniel Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011.

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 (EMNLP)*, pages 25961–25970. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1318. URL https://aclanthology.org/2025.emnlp-main.1318/.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, 2020.

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 (NeurIPS)*, 2020.

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 (TACL)*, 12:157–173, 2024.

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

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

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

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. *International Conference on Learning Representations (ICLR)*, 2025.

Zhaofen Wu, Hanrong Zhang, Fulin Lin, Wujiang Xu, Xinran Xu, Yankai Chen, Henry Peng Zou, Shaowen Chen, Weizhi Zhang, Xue Liu, Philip S. Yu, and Hongwei Wang. GAM: Hierarchical graph-based agentic memory for LLM agents. *arXiv preprint arXiv:2604.12285*, 2026.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2024. Introduces the BGE embedding models, incl. `bge-small-en-v1.5`; arXiv:2309.07597.

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, 2025. doi: 10.52202/085713-0593.

A PROMPTS

For auditability we reproduce the evaluation-facing prompts from the repository; repeated judge boilerplate is shown once and non-ASCII characters are normalised for typesetting. The *answerer* system prompt (used with `--reasoning`; it instructs evidence aggregation, most-recent-wins on conflicts, and abstention only as a last resort):

```
You answer the user's question using the provided dated context. The context has two parts: a
      FACTS
index (a digest) AND the full dated CONVERSATIONS below it. The answer is USUALLY present --
     search BOTH
parts thoroughly before concluding anything. The dates are real; use them for temporal
     reasoning.

DIG FOR THE ANSWER FIRST. Most questions ARE answerable from the history -- your job is to
     find the
evidence, not to give up. If the FACTS digest doesn't contain it, scan the full CONVERSATIONS
      below.

WHEN THE QUESTION NEEDS MULTIPLE EVIDENCE PIECES (counting, summing, ordering, date
     arithmetic, duration,
earliest/latest/most-recent, multi-step lookup, knowledge updated over time):
  EVIDENCE: list EVERY relevant dated item from the context, one per line with its date. Be
     EXHAUSTIVE
  -- scan the whole history; a missed item makes a count or total wrong. Re-read before
     finalizing.
  REASON: count distinct items / sum / sort by date / compute the date difference / pick the
     most recent.
  Show the work in 1-2 lines. For durations compute end_date - start_date explicitly.
  ANSWER: <a single concise final answer -- no reasoning on this line>

FOR DATE/TIME/NUMBER ANSWERS: give the MOST SPECIFIC value the context supports (exact date >
      month+year
> year; exact duration). Read the exact figure from the text -- don't round (e.g. 27 minutes
     45 seconds,
not 28 minutes).

WHEN THE QUESTION ASKS FOR A RECOMMENDATION / PREFERENCE: do NOT refuse, do NOT ask a
     clarifying question,
do NOT give generic advice. Ground the answer in the user's OWN stated history: name the
     SPECIFIC people,
places, brands, tools, experiences or constraints they mentioned, and tailor the suggestion
     to those.

FOR SIMPLE SINGLE-FACT QUESTIONS: go straight to 'ANSWER: <fact>'.

CURRENT-STATE questions ('what is my current/latest X', 'where do I work now'): report ONLY
     the most
recent value by date and explicitly disregard older, superseded ones. The newest dated
     statement wins.
When facts conflict, ALWAYS trust the most recent one by date.

ABSTENTION -- a careful LAST resort, only after searching the ENTIRE history (facts AND all
     conversations)
and the SPECIFIC thing asked is genuinely never stated. Do NOT abstain just because it wasn't
      in the FACTS
digest. If the question presupposes something the user truly never mentioned, reply exactly '
     ANSWER: I
don't know' rather than guessing. Never fabricate a value. The 'ANSWER:' line is REQUIRED.
```

The answer template (the context is the only thing that varies across systems):

```
Today's date: {qdate}

{context}

Question: {question}
Answer:
```

The System-2 fact-extractor system prompt (a non-English worked example is elided for typesetting; values are kept in the user's original language, only the predicate is English snake_case):

```
You are a precise information-extraction engine for a long-term memory system. From a multi-
     turn
conversation, extract the atomic, durable facts it states about the user and the people/
     things they
```

```
mention (identities, attributes, preferences, relationships, possessions, goals/plans, and
     events with
their times). Output ONLY a JSON array of objects, each with keys "subject", "predicate", "
     object",
"text". Use short snake_case predicates (e.g. works_at, lives_in, favorite_color, owns,
     married_to,
born_in, visited). Capture PREFERENCES explicitly with predicates like likes, dislikes,
     prefers, avoids,
allergic_to, favorite_<thing>; for each preference or dislike stated, output a SEPARATE fact.
      Resolve
first-person ('I','my','me') to the user's name when known, otherwise to "user". Capture a
     stated name as
{"subject":"user","predicate":"name","object":"<Name>"}. LANGUAGE: keep subject and object
     VALUES (names,
places, brands, free text) in the SAME language the user used -- do NOT translate them; only
     the predicate
stays English snake_case. "text" is a natural one-sentence statement in the conversation's
     language. Do
NOT infer or invent facts that are not stated. If there are no durable facts, output [].
```

We grade with the *official* LongMemEval category-specific judge prompts. The common instruction is shown once below, followed by each category-specific addition:

```
[single-session-user / single-session-assistant / multi-session]
I will give you a question, a correct answer, and a response from a model. Please answer yes
     if the
response contains the correct answer. Otherwise, answer no. If the response is equivalent to
     the correct
answer or contains all the intermediate steps to get the correct answer, you should also
     answer yes. If
the response only contains a subset of the information required by the answer, answer no.
Question: {q}   Correct Answer: {a}   Model Response: {r}
Is the model response correct? Answer yes or no only.

[temporal-reasoning]  (as above, plus:)
... do not penalize off-by-one errors for the number of days. If the question asks for the
     number of
days/weeks/months and the model makes off-by-one errors (e.g., predicting 19 days when the
     answer is 18),
the model's response is still correct.

[knowledge-update]  (as above, plus:)
... If the response contains some previous information along with an updated answer, the
     response should
be considered correct as long as the updated answer is the required answer.

[single-session-preference]
I will give you a question, a rubric for desired personalized response, and a response.
     Answer yes if the
response satisfies the desired response. The model need not reflect all points in the rubric;
      the response
is correct as long as it recalls and utilizes the user's personal information correctly.

[abstention / unanswerable]
I will give you an unanswerable question, an explanation, and a response. Answer yes if the
     model correctly
identifies the question as unanswerable (it may say the information is incomplete, or give
     other
information but not the asked information).
```

B Evaluation Controls

The final harness enforces the following controls uniformly across every reported system:

1. **Context isolation.** The final system answers only from its retrieved slice; full-context receives the complete history; 8k-budget RAG and ENGRAM share the same per-question token cap.
2. **Frozen random boundaries.** Contexts are materialized once and reused across readers, judges, and repeated generations. Changing a judge cannot trigger retrieval or answer regeneration.
3. **Official grading semantics.** Every judge uses the LongMemEval category-specific prompts, including the official temporal, knowledge-update, preference, and unanswerable rules.

4. **Paired comparisons.** Systems share question IDs, answer prompts, reader settings, and judge settings; single-variable ablations inherit the final configuration and change one mechanism only.
5. **Artifact identity.** Dataset revision, code tree, prompts, models, dependencies, and configuration are fingerprinted in the manifest. Each context, answer, and judgment has an immutable key.