When Relevant Memories Are Wrong: A Controlled Reliability Audit of Episodic Memory Retrieval

Anonymous authors
Paper under double-blind review

Abstract

Episodic memory systems retrieve prior episodes by semantic similarity, but relevance does not guarantee a correct outcome. We construct paired GSM8K memories that preserve the task and most of the worked derivation while changing the final conclusion, and evaluate retrieval and downstream use under matched memory contexts. Across all 1,319 test questions, Qwen3-Embedding-4B ranks the verified episode first in 40.3% of final-only pairs and 66.4% of structured pairs; BGE-M3 reaches 32.1% and 39.8%, respectively. For BGE-M3, paraphrasing changes selection by at most 0.2 percentage points, whereas removing the duplicated question changes Good@1 by 9.3–12.6 percentage points. In a frozen response bank, verified memory raises accuracy from 93.0% to 99.0%, while corrupted memory reduces it to 56.5%. Semantic Top-1 reaches 81.0% versus 77.5% for matched random selection; the paired 95% CI includes zero. Natural incorrect-solution and temporal-evidence evaluations show that retrieval relevance, episode correctness, and generator utilization should be reported as distinct quantities.

1 Introduction

Language agents store prior interactions, reflections, and solutions in external memory. A retriever encodes the current query, ranks stored episodes, and places selected text in the generator context. This pattern supports long-term conversational memory (Zhong et al., 2024; Packer et al., 2023), simulated agents (Park et al., 2023), and reflective learning (Shinn et al., 2023) with transformer language models (Vaswani et al., 2017; Brown et al., 2020).

Each episode is associated with the current query and a recorded outcome. Semantic similarity estimates the relation from text, whereas outcome correctness requires verification. A stored solution can match the query closely and still contain an incorrect conclusion. Retrieval-augmented generation studies examine the effect of relevant context that conflicts with the answer (Hong et al., 2024; Chen et al., 2022).

We measure the two stages at separate interfaces. The retrieval interface ranks a verified episode against a near-duplicate episode with a changed outcome. The utilization interface replays the same question under fixed memory contexts and measures the generated answer. Shared episodes and matched model evaluations connect the retrieval decision to the downstream response.

An end-task score alone cannot localize failure to retrieval, episode correctness, or generator behavior, although these failure modes require different remedies. We therefore ask whether a retriever can distinguish similarly relevant episodes with different outcomes and how that selection propagates under a matched context budget.

The primary audit uses GSM8K (Cobbe et al., 2021). For every one of its 1,319 test questions, we construct a verified episode from the official worked solution and a paired corrupted episode by changing the final numeric conclusion while retaining the question and most of the derivation. A frozen 100-question response bank replays five ordered contexts per question: no memory, each singleton episode, and both episodes in opposite orders. The same bank supports semantic Top-1 and equal-budget random Top-1 through stored singleton responses. Naturally written incorrect solutions and LongMemEval-S extend the audit to a different error surface and to temporal evidence.

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.

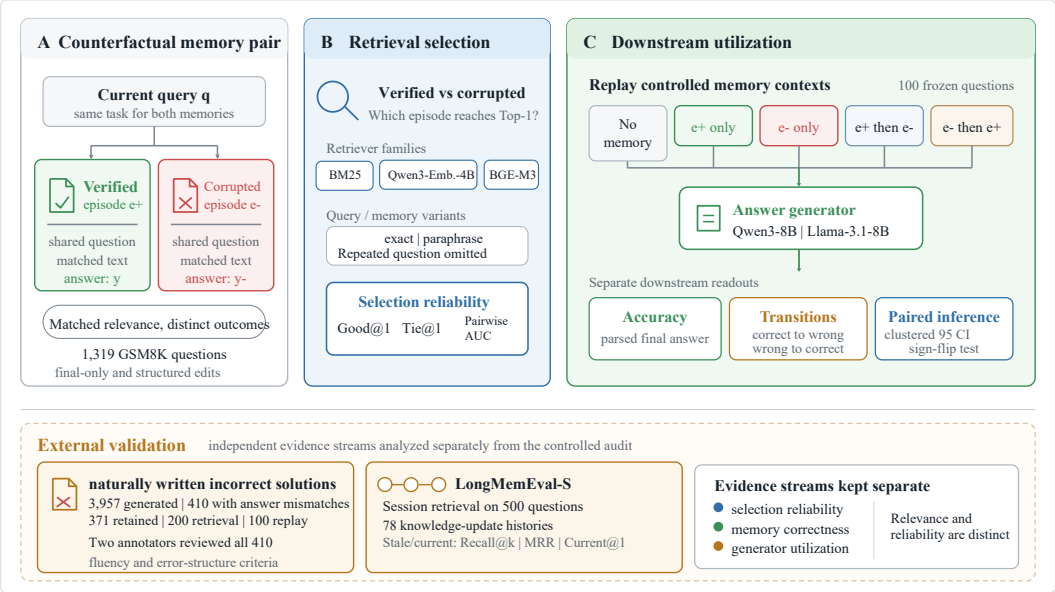


Figure 1: Audit framework for episodic memory retrieval. Counterfactual memory pairs preserve the task and most of the worked solution while changing the final conclusion from verified e^+ to corrupted e^-. Retrieval is evaluated as a selection problem, while downstream behavior is replayed under five controlled memory contexts. The lower band reports naturally written incorrect solutions and LongMemEval-S as separate evidence streams for episode selection, outcome correctness, and generator utilization.

Our contributions are:

- a pairwise retrieval audit with explicit Good@1, Tie@1, and Pairwise AUC definitions for verified-versus-corrupted episode selection;
- a frozen response-bank protocol that holds model evaluations fixed while varying memory content, retrieval policy, and context order; and
- an evaluation across deterministic edits, naturally written incorrect solutions, and temporal evidence updates, with paired uncertainty estimates and policy replay.

2 Related Work

Memory for language agents. MemoryBank stores and retrieves long-term conversational information (Zhong et al., 2024), while MemGPT treats context management as a virtual-memory problem (Packer et al., 2023). Generative Agents retrieve observations and synthesize reflections to support planning and social behavior (Park et al., 2023). Reflexion places verbal feedback in an episodic buffer for later trials (Shinn et al., 2023). These systems demonstrate that textual memory can extend agent behavior. The quality of stored and retrieved text is part of inference. Recent benchmarks evaluate long-term conversational memory across multi-session and temporal reasoning tasks (LoCoMo) (Maharana et al., 2024), information extraction, multi-session reasoning, knowledge updates, and abstention (LongMemEval) (Wu et al., 2024), and incremental interactions across sessions (MemoryAgentBench) (Hu et al., 2026). These benchmarks measure task-level memory performance. Our audit measures selection between episodes with matched semantic content and the downstream effect of the selected episode.

Retrieval and unreliable context. Retrieval-augmented generation conditions a model on selected external passages (Lewis et al., 2020); dense passage retrieval provides a standard dual-encoder baseline for selecting such passages (Karpukhin et al., 2020). Counterfactual noise makes this selection problem explicit: a document can be query-relevant and misleading at the same time (Hong et al., 2024). Distracting-context studies show that passage effects depend on the query and

the generator (Amiraz et al., 2025). Utility-oriented retrieval evaluation also separates human relevance from value to an LLM consumer (Trappolini et al., 2026). Adaptive retrieval and self-critique methods such as SELF-RAG model retrieval and evidence assessment as separate decisions (Asai et al., 2024); credibility-aware generation explicitly models the trustworthiness of retrieved contexts (Pan et al., 2024). We apply these distinctions to episodic memory with fixed verified, corrupted, and conflicting contexts.

Retrieval-augmented language models scale access to external knowledge (Borgeaud et al., 2022), and tool-using agents interleave reasoning with environment actions (Yao et al., 2023; Schick et al., 2023). Our audit measures the memory-selection action before generation. Chain-of-thought and self-consistency provide controls for downstream variation (Wei et al., 2022; Wang et al., 2023).

The distinction between retrieval and utilization is also important for evaluation realism. DRUID shows that synthetic context-utilization tests can misrepresent the complexity and source variability of real retrieved evidence (Hagström et al., 2025). Our synthetic near-duplicate audit measures selection under controlled counterfactuals, while the LongMemEval-S analysis measures temporal evidence selection and direct context utilization.

Difference from standard negative retrieval. Standard dense retrieval ranks gold evidence against non-gold candidates (Karpukhin et al., 2020). Our negative is a matched counterfactual rather than an irrelevant passage: it preserves the query and most of the derivation while changing the final conclusion. Retrieval therefore tests outcome correctness within a fixed topic, while replay measures downstream use.

3 Problem Formulation

Let a query q have numeric answer y, verified episode e^+, and corrupted episode e^-. A semantic encoder ϕ gives retrieval scores

$$s^+ = \cos(\phi(q), \phi(e^+)), \qquad s^- = \cos(\phi(q), \phi(e^-)).$$

For strict retrieval reporting, Good@1 counts a query when $s^+ > s^-$. Exact equality is reported as Tie@1, and Incorrect@1 counts a query when $s^+ < s^-$. We define

$$\text{Good@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[s_i^+ > s_i^-], \qquad \text{Tie@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[s_i^+ = s_i^-],$$

$$\text{Incorrect@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[s_i^+ < s_i^-].$$

Tables may abbreviate Tie@1 as "Tie". Pairwise AUC is the probability that a random query ranks the verified episode above the corrupted episode, with half credit for an exact tie:

$$\text{Pairwise AUC} = \frac{1}{N}\sum_{i=1}^{N} \left(\mathbb{I}[s_i^+ > s_i^-] + \tfrac{1}{2}\mathbb{I}[s_i^+ = s_i^-]\right) = \text{Good@1} + \tfrac{1}{2}\text{Tie@1}.$$

With one verified and one corrupted episode per query, $\text{Good@1} + \text{Tie@1} + \text{Incorrect@1} = 1$. A random ordering has expected Pairwise AUC 0.5. Tests against this reference use Pairwise AUC; ties contribute half credit to Pairwise AUC and zero credit to strict Good@1.

Retrieval supplies context for the final task. Let $f(q, S; \xi)$ be the parsed answer from generation seed ξ under ordered memory context S. We store

$$S \in \{\varnothing, [e^+], [e^-], [e^+, e^-], [e^-, e^+]\}.$$

This response bank supports verified-only, corrupted-only, both orderings, semantic Top-1, and random Top-1 policies. Semantic and random policies choose between the already stored singleton responses and use no additional model samples.

The audit addresses four questions:

1. the frequency with which semantic similarity ranks the verified episode above its near-duplicate corruption;

2. the accuracy change produced by verified and corrupted singleton memories relative to no memory;
3. generator recovery and order sensitivity when both episodes are present; and
4. the accuracy difference between semantic Top-1 and equal-budget random Top-1.

4 Counterfactual Memory Construction

4.1 Frozen Questions

We use the GSM8K test set, a benchmark of grade-school mathematical word problems with worked solutions (Cobbe et al., 2021). We retain items with a numeric answer marker, compute a SHA-256 identifier from the exact question string, sort by that identifier, and select the first 100. This content-based rule avoids choosing a convenient contiguous slice. We verify the source and selected items with cryptographic hashes.

4.2 Verified and Corrupted Episodes

The verified episode concatenates the question and official worked solution. We evaluate two corruption variants. In the final-only variant, we change only the final answer marker. In the structured variant, we also change matching occurrences on the immediately preceding reasoning line. In both variants, the final answer is changed by

$$\delta(y) = \max(1, \mathrm{round}(0.1|y|)), \quad y^- = y + \delta(y).$$

The final answer marker is replaced with y^-. All other text is held fixed.

Both variants create near-duplicate negative episodes while preserving the original derivation. In the final-only variant, the changed conclusion can create an arithmetic inconsistency visible to the generator; the structured variant removes the matching intermediate occurrence as well. These variants keep retrieval similarity high while separating conclusion-only from structured corruption.

4.3 Five Context Interventions

Each item receives five independent model evaluations per seed. The no-memory prompt contains only the current question. Singleton prompts contain either e^+ or e^-. Conflict prompts contain both memories in opposite orders. All prompts state that retrieved memories are past worked examples that may be imperfect and should be used only if helpful. This wording is fixed across conditions.

The two orderings identify recency or position sensitivity. A difference between $[e^+, e^-]$ and $[e^-, e^+]$ measures an ordering effect that an order-invariant analysis would omit. Semantic and random Top-1 reuse their chosen singleton contexts.

5 Policies

No memory selects $f(q, \varnothing)$. **Verified oracle** selects $f(q, [e^+])$ and represents the verified-memory reference for this two-item store. **Corrupted only** selects $f(q, [e^-])$ and measures exposure to the corrupted episode. **Verified-first** and **Corrupted-first** select the two conflict orderings. **Semantic Top-1** selects the singleton response associated with the larger cosine score. **Random Top-1** uses a frozen policy RNG to select one singleton with probability $1/2$. It matches semantic Top-1 in memory count and evaluation budget.

The oracle uses the known provenance label and serves as a reliability reference for this two-item store. Corrupted-only serves as a corrupted-memory control. The primary policy comparison is semantic versus random Top-1.

6 Experimental Setup

Models and decoding. We use qwen3-embedding-4b for memory embeddings and Qwen3-8B (Yang et al., 2025) for generation using Ollama. Qwen3-8B reasoning mode is disabled. For the full-

Retriever	Corruption	Good@1	Tie@1	Pairwise AUC (95% CI)
BM25	Final-only	3.6	93.9	50.5 [49.89,51.21]
BM25	Structured	3.8	93.6	50.6 [49.89,51.25]
Qwen3-Emb.	Final-only	40.3	0.0	40.3 [37.7,42.9]
Qwen3-Emb.	Structured	66.4	0.0	66.4 [63.8,68.9]
BGE-M3	Final-only	32.1	0.3	32.3 [29.8,34.8]
BGE-M3	Structured	39.8	0.2	39.9 [37.3,42.5]

Table 1: Full GSM8K retrieval audit on 1,319 paired memories. Good@1 is strict selection of the verified episode; Tie@1 is reported separately, and Pairwise AUC assigns half credit to an exact tie. The BM25 and BGE-M3 intervals use question-level bootstrap; the no-tie Qwen3 intervals use Wilson intervals.

split BGE-M3 audit, we use the local encoder with float32 weights, batch size 32, and maximum sequence length 256. Generation uses temperature 0.3, a 384-token cap, and base seeds 202741 and 202742. Each question–context pair receives a derived deterministic evaluation seed. The second run reuses the exact frozen episodes and embeddings, changing only generation seeds.

Parsing. The prompt requests a final answer line. The parser takes the last explicit numeric marker, removes comma separators, and compares floating values after rounding to six decimal places. A missing or malformed marker is an abstention and is scored incorrect. Generated text is retained for independent checks of parsing and contamination.

Cost. We report prompt tokens, generated tokens, and their sum for the selected response of each policy. Verified, corrupted, semantic, and random Top-1 each inject exactly one memory but may differ slightly in tokens because the corrupted number can have a different length. Conflict policies inject two episodes. Experimental precomputation requires 500 model evaluations per seed; each policy uses one selected response per item.

Inference. The two generation runs share 100 questions. The question identifier defines the independent cluster; the 200 item–seed rows are repeated measurements. For a paired policy contrast, we average the two within-question correctness differences, sample 100 clusters with replacement 100,000 times for a percentile interval, and use a two-sided cluster sign-flip test with 100,000 draws. The analysis uses the frozen question clusters as the unit of inference.

Audit. The audit verifies 100 unique content identifiers, five evaluations per item, distinct verified and corrupted conclusions, complete token fields, parser outcomes, semantic and random selections, and policy assignments. We recompute all reported quantities from the underlying observations and verify that the same question set is used across seeds before clustering.

7 Results

7.1 Full-Split Retrieval Audit

We evaluate the complete 1,319-question GSM8K test split with two paired corruption mechanisms. Table 1 reports strict Good@1, Tie@1, and Pairwise AUC as defined in Section 3. BM25 is a lexical retrieval baseline based on the probabilistic relevance framework (Robertson & Zaragoza, 2009); it also provides a tie control because the question text is identical in both memories. Qwen3-Embedding-4B and BGE-M3 (Chen et al., 2024) show corruption-dependent selection. The full-split BGE-M3 results have question-level bootstrap 95% Pairwise AUC intervals of $[29.8, 34.8]$ for final-only and $[37.3, 42.5]$ for structured edits.

7.2 Query Formulation and Question Duplication

We conduct a separate BGE-M3 query ablation on the same 1,319-question set under three query/memory conditions: the exact question with the full memory, an independently produced paraphrase with the full memory, and the exact question with the repeated question removed from each memory. The paraphrases were generated without access to candidate memories or gold answers and were validated for preserved numbers, entities, relations, and question goals. Table 2 reports

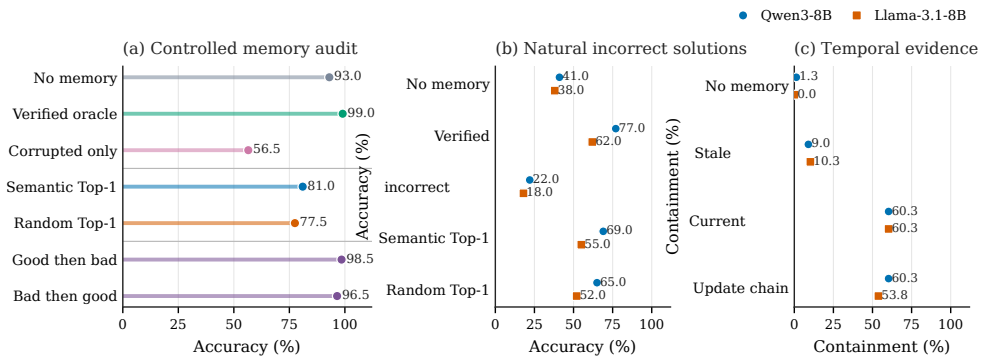


Figure 2: Downstream memory utilization across three settings. Panel (a) reports the frozen synthetic response-bank audit as a horizontal point plot; panel (b) reports accuracy on naturally written incorrect memories for both generators; panel (c) reports answer-string containment after direct injection of stale or current LongMemEval-S evidence. Panels (a) and (b) use answer accuracy, whereas panel (c) uses answer-string containment. Marker shape and color identify the generator in panels (b) and (c).

the absolute selection rates and within-run condition differences for this ablation. The exact-query/full-memory condition is the ablation control; the full-split estimate in Table 1 comes from the independent full-split run.

Condition	Final-only Good@1 / Pairwise AUC (95% CI)	Structured Good@1 / Pairwise AUC (95% CI)
Exact query + full memory	31.5 / 31.5 [29.0,34.0]	36.0 / 36.0 [33.4,38.6]
Paraphrase + full memory	31.6 / 31.6 [29.1,34.1]	36.2 / 36.2 [33.6,38.7]
Exact query + question-omitted memory	44.1 / 44.1 [41.5,46.8]	45.3 / 45.3 [42.7,48.0]
Paraphrase − exact (pp)	+0.1 [-1.4,1.6]	+0.2 [-1.4,1.7]
Question-omitted − exact (pp)	+12.6 [9.3,15.8]	+9.3 [6.4,12.3]

Table 2: BGE-M3 query and question-duplication ablation. Each absolute entry is reported as Good@1 / Pairwise AUC in percent, followed by the item-level bootstrap 95% CI for Pairwise AUC. Pairwise AUC equals Good@1 in every condition because no exact ties occurred. Difference rows report paired percentage-point changes and paired 95% CIs computed within the ablation run on the 1,319-item question set.

Within the ablation run, paraphrasing changes Good@1 by 0.1–0.2 percentage points; the paired 95% CIs include zero in both corruption conditions. Removing the duplicated question changes Good@1 by 9.3–12.6 percentage points, with the question-omitted conditions at 44.1–45.3%. The corrupted episode remains the rank-1 selection in more than half of the question-omitted pairs.

Table 1 reports the Pairwise AUC values: BGE-M3 remains below chance for final-only and structured edits (32.3% and 39.9%), whereas Qwen3-Embedding-4B improves from 40.3% to 66.4%; the plot and question-level bootstrap intervals are provided in Appendix A.2.

7.3 Controlled Response-Bank Replay

On the frozen 100-question response bank, Table 3 combines two generation seeds. No memory obtains 186/200 (93.0%). Verified memory changes 12 no-memory errors to correct answers and reaches 198/200 (99.0%), with no correct-to-incorrect transition in the singleton responses. Its paired gain is $+6.0$ percentage points with question-clustered 95% CI $[2.0, 11.0]$ and sign-flip $p = .0159$.

Corrupted memory obtains only 113/200 (56.5%), a -36.5 percentage-point difference from no memory (95% CI $[-47.0, -26.0]$, $p < .0001$). Across the two runs, the generated answer exactly repeats the injected wrong final answer in 82/200 corrupted contexts. The transition pattern is directional: 82 outcomes change from a correct no-memory answer to an incorrect corrupted-memory answer, while nine move in the opposite direction.

Policy	Correct	Acc.	Tokens	Memory
No memory	186/200	93.0	263	0
Verified oracle	198/200	99.0	411	1
Corrupted only	113/200	56.5	411	1
Verified-first	197/200	98.5	609	2
Corrupted-first	193/200	96.5	609	2
Semantic Top-1	162/200	81.0	413	1
Random Top-1	155/200	77.5	411	1

Table 3: Combined results over 100 shared questions and two generation seeds. Tokens are prompt plus generated tokens for the selected response; Memory is the number of injected episodes.

(a) Paired contrasts

Contrast	Δ pp	Cluster 95% CI	p
Semantic – random	+3.5	$[-3.0, 10.5]$	.384
Semantic – verified	-18.0	$[-25.5, -11.0]$	$< .0001$
Semantic – corrupted	+24.5	$[16.5, 33.0]$	$< .0001$
Verified – no memory	+6.0	$[2.0, 11.0]$	.0159
Corrupted – no memory	-36.5	$[-47.0, -26.0]$	$< .0001$
Verified-first – corrupted-first	+2.0	$[0.0, 4.5]$	.250

(b) Correctness transitions

First vs. second	0/0	0/1	1/0	1/1
Corrupted vs. no memory	5	82	9	104
Verified vs. no memory	2	0	12	186
Semantic vs. random	23	15	22	140
Verified-first vs. corrupted-first	3	0	4	193

(c) Paired win/tie/loss

Contrast	Wins	Ties	Losses
Verified − no memory	12	188	0
Corrupted − no memory	9	109	82
Semantic − random	22	163	15
Verified-first − corrupted-first	4	196	0

Table 4: Paired evidence for the frozen response-bank replay over 200 item–seed rows. Panel (a) reports cluster-bootstrap contrasts and paired sign-flip tests with 95% intervals; panel (b) gives the complete correctness transition matrix; panel (c) reports the same paired comparisons as win/tie/loss counts.

7.4 Semantic Top-1 and Matched Random Retrieval

Semantic Top-1 obtains 162/200 (81.0%) and random Top-1 obtains 155/200 (77.5%). Their 3.5 percentage-point difference has clustered 95% CI $[-3.0, 10.5]$ and $p = .384$. The methods agree on correctness in 163 of 200 item–seed rows; 22 discordances favor semantic and 15 favor random. The paired interval includes zero.

The verified oracle is 18.0 percentage points higher than semantic Top-1 (95% CI $[11.0, 25.5]$, $p < .0001$). Semantic Top-1 is 24.5 percentage points higher than the corrupted-only policy (95% CI $[16.5, 33.0]$, $p < .0001$).

Effect estimates quantify the magnitude of each policy difference, while outcome transitions identify which errors produce that difference. Table 4 presents both outputs with the corresponding win/tie/loss counts in three panels. The semantic–random comparison has 22 wins and 15 losses for semantic retrieval; its clustered 95% CI is $[-3.0, 10.5]$ percentage points. In contrast, corrupted memory changes 82 correct no-memory outcomes to errors and repairs only nine.

7.5 Conflicting Evidence Enables Recovery

When both memories are present, accuracy returns near the verified oracle: 197/200 with the verified episode first and the corrupted episode second, versus 193/200 in the opposite order. The two order-

Generator	No mem.	Verified	Corrupted	Semantic	Random
Qwen3-8B	91.0	98.0	73.0	89.0	84.3
Llama-3.1-8B	69.0	97.0	90.0	98.0	95.7

Table 5: Cross-generator replay accuracy (percent). Each cell contains 100 questions and three generation seeds.

(a) Retrieval on 200 naturally written incorrect pairs

Retriever	Good@1	Tie@1	Incorrect@1	Pairwise AUC
BM25	46.5	0.0	53.5	46.5
BGE-M3	37.0	0.0	63.0	37.0
Qwen3-Embedding-4B	76.0	0.0	24.0	76.0

(b) Downstream replay accuracy (percent)

Generator	No mem.	Verified	Naturally incorrect	Semantic	Random	Verified-first	Incorrect-first
Qwen3-8B	41.0	77.0	22.0	69.0	65.0	43.0	42.0
Llama-3.1-8B	38.0	62.0	18.0	55.0	52.0	32.0	53.0

Table 6: Naturally written incorrect-solution audit. Panel (a) reports retrieval on 200 verified/incorrect pairs. Good@1 selects the verified episode, Tie@1 records an exact score tie, Incorrect@1 selects the naturally written incorrect episode, and Pairwise AUC assigns half credit to a tie. Panel (b) reports replay accuracy on 100 questions with three generation seeds; the last two columns show the verified-first and incorrect-first conflict orderings.

ings differ by four outcomes, all favoring verified-first. The clustered contrast is $+2.0$ percentage points with a 95% CI of $[0.0, 4.5]$ and $p = .250$.

The two-memory condition allows the generator to compare the verified and corrupted episodes; the singleton condition removes that comparison. Presenting the memories in opposite orders also measures position sensitivity in long contexts (Liu et al., 2024). In the Qwen3-8B response bank, corrupted memory alone reduces accuracy, whereas the two-memory condition usually yields a correct answer when the verified reasoning is also visible. A Top-1 retriever removes that comparison by discarding one episode. These conditions quantify the trade-off between context cost and access to conflicting evidence.

7.6 Cross-Generator Replay

We replay the same 100 frozen synthetic pairs and five context prompts with Qwen3-8B and Llama-3.1-8B, using three generation seeds per model. Verified memory improves over no memory for both generators, but the effect of a corrupted memory changes sign. Table 5 reports the replay accuracy for each generator and policy.

Verified memory improves by 7 percentage points for Qwen3 (95% CI $[3, 13]$) and by 28 percentage points for Llama (95% CI $[19, 37]$). Corrupted memory lowers Qwen3 by 18 percentage points (95% CI $[-28, -8]$) but raises Llama by 21 percentage points (95% CI $[11, 31]$). Semantic Top-1 and matched random obtain 89.0% and 84.3% for Qwen3-8B, and 98.0% and 95.7% for Llama-3.1-8B. The corresponding paired intervals are $[-2, 14]$ and $[0, 7]$ percentage points, respectively. The effect of corrupted memory is therefore generator dependent.

7.7 Naturally Written Incorrect Solutions

We extend the audit beyond a final-number replacement with 3,957 candidate solutions. Of these, 410 have a final answer different from the GSM8K gold answer, and an independent model validator retained 371. Two human annotators independently reviewed all 410 candidates with a fixed rubric covering grammatical fluency, naturalness of the error, plausible error structure, and absence of obvious edit artifacts. Each candidate was checked against the problem and gold answer with retrieval and downstream results hidden; disagreements were resolved before the pool was frozen. We froze 200 items for retrieval and 100 for downstream replay; the two analyses use their respective frozen item sets.

Panel (a) of Table 6 gives the retrieval results on naturally written incorrect solutions. BGE-M3 selects the verified episode on 37.0% of pairs, while Qwen3-Embedding-4B selects it on 76.0%. Panel (b) gives the replay results. Naturally written incorrect memory lowers accuracy relative to no mem-

ory by 19 percentage points for Qwen3 (95% CI $[-28, -11]$) and 20 percentage points for Llama (95% CI $[-30, -10]$). Verified memory improves accuracy by 36 and 24 percentage points, respectively. Semantic Top-1 differs from random by 4 percentage points for Qwen3 (95% CI $[-5, 13]$) and 3 percentage points for Llama (95% CI $[-5, 11]$). Panel (b) of Figure 2 places these results beside the controlled response-bank results for the same generators.

7.8 LongMemEval-S Temporal Retrieval and Utilization

We next evaluate retrieval on the complete cleaned LongMemEval-S release (Wu et al., 2024): all 500 questions, with candidates restricted to each question's own chronological history. Every candidate is a session-level unit; round-level units are excluded. The gold unit is the official answer-bearing session. For the 78 knowledge-update questions, Current@1 is the fraction of queries whose later answer-bearing session ranks first.

BGE-M3 obtains the highest overall Recall@1 among the reported retrievers. BM25 ranks the later answer-bearing session first more often on the knowledge-update subset; BGE-M3 ranks it first on 50.0% of those questions. Overall evidence recall and temporal selection are reported separately. The complete retrieval metrics are reported in Table 7; candidate definitions, denominators, and backend composition are documented in Appendix A.1.

8 Limitations

The controlled GSM8K pairs are deliberately difficult near-duplicates. They hold most text fixed under final-only and structured edits; naturally written memory histories are evaluated separately. The naturally written incorrect- solution audit begins with model-generated candidates and applies a two-annotator human review. The review was adjudicated before the pool was frozen, so it defines the inclusion protocol but does not provide an independent inter-annotator agreement estimate. Its 200-item sample describes that audit distribution; deployment prevalence requires a natural-history sample.

The downstream response bank contains 100 frozen questions. Multiple seeds provide repeated generation measurements, with all paired inference clustered by question. The cross-generator and naturally written incorrect-solution replays cover two generators and the reported prompting policies. The LongMemEval-S appendix table reports retrieval on all 500 questions and direct stale/current context injection on 78 knowledge-update histories. The BGE-M3 evaluation uses the same encoder across the audit, with a local implementation used when the embedding service was unavailable.

9 Conclusion

This study measures episodic memory at three interfaces: retrieval selection, episode correctness, and generator utilization. The paired GSM8K audit uses a shared unit of one query, one verified episode, and one matched corrupted episode. Across the full test split, verified-episode selection changes with the retriever and the corruption mechanism. In the BGE-M3 ablation, paraphrasing changes Good@1 by 0.1–0.2 percentage points, while removing the repeated question changes it by 9.3–12.6 percentage points and yields 44.1–45.3% Good@1.

The frozen response bank measures the downstream effect of that selection. Verified memory raises accuracy from 93.0% to 99.0%; corrupted memory changes 82 correct no-memory outcomes into errors; and presenting both episodes gives the generator access to both conclusions. The naturally written incorrect- solution audit shows a similar retrieval contrast under a different error surface. On LongMemEval-S, BGE-M3 reaches 82.4% Recall@1 overall and 50.0% Current@1 on knowledge-update questions. Direct current-evidence injection raises containment by 51.3 percentage points for Qwen3-8B and 50.0 percentage points for Llama-3.1-8B. Together, these results support reporting retrieval selection, episode correctness, temporal evidence selection, and utilization as separate quantities in episodic-memory evaluation.

Ethics Statement

The study uses public mathematical word problems and collects no human-subject data. Two annotators reviewed generated solutions as described in the natural-solution audit; no personal interaction data were used. The synthetic corruption mechanism is for robustness evaluation only.

Reproducibility Statement

The appendix specifies the frozen items, memory construction, response bank, seeds, audits, and analysis procedure. Supplementary materials provide the response-bank data, the query-ablation summary, the question set, and the information needed to reproduce the reported analyses. The full-split BGE-M3, naturally written incorrect-solution, and LongMemEval evaluations are described by their source datasets and preprocessing procedures.

AI Use Statement

Generative AI tools were used for language editing and LaTeX reformatting. They were not used to generate or alter reported data, statistical analyses, or quantitative results. The authors reviewed the final manuscript and remain responsible for all claims, citations, and artifacts.

REFERENCES

Chen Amiraz, Florin Cuconasu, Simone Filice, and Zohar Karnin. The distracting effect: Understanding irrelevant passages in RAG. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18228–18258. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.892.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. SELF-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

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 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.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Hung-Ting Chen, Michael Zhang, and Eunsol Choi. Rich knowledge sources bring complex knowledge conflicts: Recalibrating models to reflect conflicting evidence. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 2292–2307. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.146.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. BGE M3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*, 2024. URL https://arxiv.org/abs/2402.03216.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Lovisa Hagström, Sara Vera Marjanovic, Haeun Yu, Arnav Arora, Christina Lioma, Maria Maistro, Pepa Atanasova, and Isabelle Augenstein. A reality check on context utilisation for retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19691–19730. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.968.

Giwon Hong, Jeonghwan Kim, Junmo Kang, Sung-Hyon Myaeng, and Joyce Jiyoung Whang. Why so gullible? enhancing the robustness of retrieval-augmented models against counterfactual noise. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2474–2495. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-naacl.159.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. In *International Conference on Learning Representations*, 2026.

Vladimir Karpukhin, Barlas Oguz, 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*, pp. 6769–6781. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.550.

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.

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.

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. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.747.

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

Ruotong Pan, Boxi Cao, Hongyu Lin, Xianpei Han, Jia Zheng, Sirui Wang, Xunliang Cai, and Le Sun. Not all contexts are equal: Teaching LLMs credibility-aware generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 19844–19863. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.1109.

Joon Sung Park, Joseph 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.

Stephen E. Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Giovanni Trappolini, Florin Cuconasu, Simone Filice, Yoelle Maarek, and Fabrizio Silvestri. Redefining retrieval evaluation in the era of LLMs. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8359–8375. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.eacl-long.391.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

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.

A Supplementary Diagnostics and Reproducibility Details

A.1 LongMemEval-S Details

The main text defines the LongMemEval-S candidate unit and the two retrieval denominators. This appendix gives the backend composition and the complete retrieval/utilization table. BGE-M3 uses the remote embedding service when available and a local Transformers implementation after service failures. Panel (a) evaluates the official answer-bearing session on all 500 questions, while Current@1 uses the 78 knowledge-update questions and the later answer-bearing session.

We separately evaluated the 78 native knowledge-update questions to measure generator utilization after direct evidence injection. Each history was answered with no memory, the earlier stale session, the later answer-bearing session, or both sessions concatenated in chronological order. Qwen3-8B and Llama-3.1-8B each used three generation seeds, yielding 936 responses per generator; question/history identity defined the bootstrap cluster. Panel (b) of Table 7 gives answer-string containment and mean prompt tokens for these interventions. The main text reports the paired containment contrasts and the behavior when both sessions are shown. Panel (c) of Figure 2 visualizes the same utilization contrast; its vertical quantity is answer-string containment, while panels (a) and (b) use answer accuracy. When both sessions are shown, the generated answer is closer to the stale answer than to the current gold answer on 74.4% of Qwen3 histories and 87.2% of Llama histories. These results characterize each generator's response to temporally conflicting evidence.

A.2 Pairwise AUC Diagnostic

Figure 3 summarizes the full-split pairwise AUC values with question-level bootstrap intervals. The two corruption mechanisms preserve most of the episode text, leaving the verified and corrupted memories close in embedding space. Pairwise AUC varies by retriever and corruption mechanism.

A.3 Seed Stability and Parser Failures

Appendix B.1 reports the per-seed results. Semantic Top-1 is exactly 81/100 in both seeds; no memory is 93/100; the verified oracle is 99/100. Corrupted-only differs by one item, and random Top-1 by one item. Both runs reuse the same 100 questions, so the results measure run stability on a shared content sample.

In the 100-question Qwen3 response bank, each seed contains one missing parsed answer among 500 evaluations. Seed 1 has one length-capped generation and seed 2 has two. We retain these responses without retrying, since a retry would give affected contexts an additional sample. Abstentions remain incorrect in all replays.

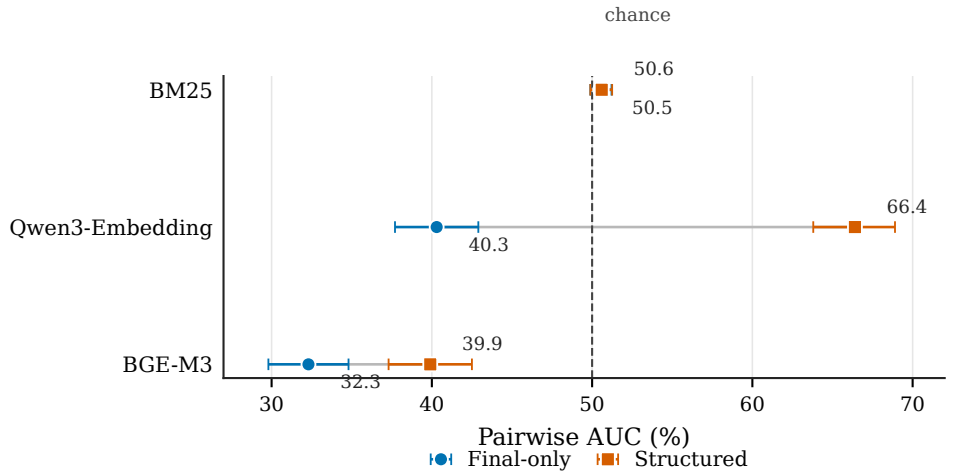


Figure 3: Pairwise retrieval AUC on the full 1,319-question GSM8K audit. Points show Pairwise AUC and horizontal whiskers show the reported 95% confidence intervals. The dashed line marks chance performance; values below it favor the corrupted episode. Structured edits improve Qwen3-Embedding-4B but leave BGE-M3 below chance.

(a) Retrieval on 500 LongMemEval-S questions

Retriever	R@1	R@5	R@10	NDCG@5	MRR	Current@1 (78 updates)
BM25	70.6	89.0	93.6	n/a	78.7	89.7
BGE-M3	82.4	96.2	98.4	87.1	88.3	50.0
Qwen3-Embedding-4B	78.4	93.6	96.8	82.7	85.5	50.0

(b) Direct stale/current utilization on 78 update histories

Condition	Qwen3 Contain.	Qwen3 Tokens	Llama Contain.	Llama Tokens
No memory	1.3	54	0.0	48
Stale evidence	9.0	3,134	10.3	3,097
Current evidence	60.3	3,058	60.3	3,028
Update chain	60.3	6,141	53.8	6,080

Table 7: LongMemEval-S retrieval and utilization. Panel (a) reports $R@k$, MRR, and NDCG@5 against the official answer-bearing session on all 500 questions; Current@1 is rank-1 selection of the later answer-bearing session on the 78 knowledge-update questions. Panel (b) reports direct context-injection results on those 78 histories; Contain. is answer-string containment and tokens are mean prompt tokens.

B Diagnostic Analysis

B.1 Seed Stability Plot

B.2 Output Responses to Retrieved Memory

The paired interventions produce three output patterns. The corrupted singleton response ends with the injected wrong number in 41/100 questions in each seed. Some corrupted contexts still yield correct answers; corrupted-only accuracy is 56.5%. Verified memory moves 12/200 outcomes from wrong to correct without a correct-to-wrong transition in the stored evaluations.

These categories are outcome based. A correct output under corrupted context can result from independent reasoning, conflict detection, or ignoring the episode. Exact reproduction of the wrong marker identifies an output-level answer-following outcome. The experiment measures output behavior; it does not measure the generator's hidden state.

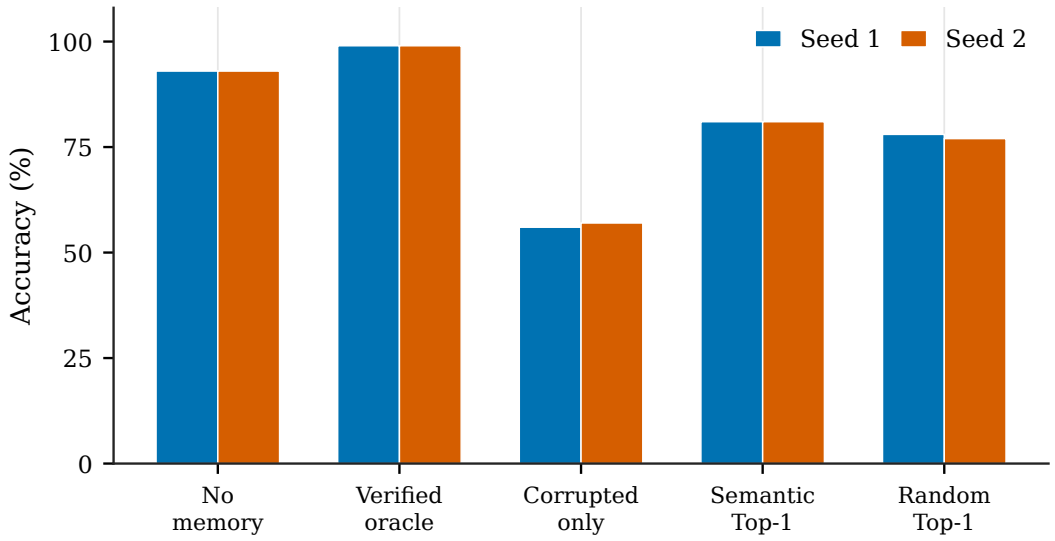


Figure 4: Accuracy by generation seed in the frozen response bank. Retrieval selections are frozen; model responses use different deterministic evaluation seeds.

Stored context	Verified ranked first	Corrupted ranked first
No memory	102/110	84/90
Verified singleton	108/110	90/90
Corrupted singleton	59/110	54/90
Verified-first	107/110	90/90
Corrupted-first	106/110	87/90

Table 8: Accuracy stratified by the frozen embedding decision. Columns contain 55 and 45 unique questions, respectively, each generated under two seeds.

B.3 Context Cost and Conflict Evidence

The verified and corrupted singleton prompts have effectively equal cost, so semantic and random policies receive the same one-memory budget after selection. The two-memory policies spend about 48% more tokens than singleton retrieval (609 versus roughly 411 per item) and recover most of the oracle performance. Top-1 uses less context and removes conflicting evidence; retrieving both enables cross-checking at a higher context cost.

A practical system can retrieve both when score margins are small, verify numeric conclusions, attach provenance, or train a utility/risk reranker. The similarity margins in this audit are small. The response bank contains no threshold-tuned abstention or multi-retrieval result.

C Retrieval–Utility Decomposition

C.1 Conditional Performance by Retrieval Decision

Table 8 partitions questions by the embedding decision. For the 55 questions where verified memory ranks first, semantic Top-1 obtains 108/110 across seeds. For the 45 questions where corrupted memory ranks first, it obtains only 54/90. In the same stratum, the verified singleton scores 90/90 and the corrupted singleton scores 54/90. The selected context is therefore associated with the observed conditional difference in this response bank.

No-memory accuracy is 92.7% when the verified episode ranks first and 93.3% when the corrupted episode ranks first. The two retrieval strata therefore have similar base-model accuracy, while semantic Top-1 replaces many correct no-memory answers with harmful context in the corrupted-selection stratum.

Retrieval is deterministic in this experiment: both generation seeds reuse the same embeddings, so the same 45 questions select corrupted memory. The downstream outcomes are stochastic but highly

Risk	Design control	Evidence retained
Selective sampling	Hash-based sampling	Source and item identifiers
Unmatched generations	Five-context bank	1,000 generated responses
Stage conflation	Exact singleton replay	Frozen question assignments
Parser recovery bias	Fixed last-marker rule	Raw and parsed outputs
Seed pseudoreplication	Cluster by question	Clustered analysis
Undocumented retries	No repair or retries	Failure status retained
Figure transcription	Programmatic plotting	Figure-generation procedure

Table 9: Integrity risks, controls, and auditable records.

stable. This separation is why we report retrieval reliability once over 100 pairs, while accuracy uses two generation runs clustered by those same pairs.

C.2 Token Accounting

No-memory responses average 263 total tokens. A singleton memory raises this to approximately 411, and two memories to 609. These totals combine prompt and generated tokens reported by Ollama. They measure context and generation length; response length varies with the generated reasoning.

Semantic and random Top-1 have the same one-memory budget by construction. Their roughly two-token mean difference arises because verified and corrupted episodes can differ in digit length and because the policies select different stored generations. The verified oracle is likewise cost-matched to corrupted only. The principal accuracy gaps therefore reflect memory content and selection under a matched one-memory budget; exact prompt lengths differ slightly.

Retrieving both memories costs about 48% more tokens than singleton retrieval but recovers 31.5–42.0 percentage points over corrupted-only and 15.5–17.5 percentage points over semantic Top-1. In this two-item stress test, these values quantify the cost and benefit of exposing the generator to both sides.

D Auditability

The protocol fixes content identifiers, corruption, embeddings, contexts, evaluation seeds, the parser, the policy RNG, and the inference procedure before combining the two formal runs.

The counterfactual construction and matched replay define the internal validity of the controlled audit. The benchmark uses deliberately controlled negatives.

E Validity Analysis

Internal validity. The response bank holds generation candidates fixed across replayed policies, and the second seed reuses byte-identical items and embeddings. Deterministic content identifiers, fixed evaluation seeds, generated responses, and exact replay constrain implementation variation. The random policy uses one frozen draw per seed, with no averaging across random retrieval assignments.

Construct validity. The corruption changes the final conclusion and, when possible, occurrences on the preceding line. It preserves the original derivation rather than regenerating a globally coherent wrong solution. Some negatives are therefore detectable from the derivation. The stress test measures reliability as agreement with the known GSM8K answer; source trust, calibration, and factual freshness are separate properties.

The exact question appears inside both memories. This creates duplicate task episodes and removes topical relevance as a differentiator. The result characterizes outcome selection under a near-duplicate challenge; ordinary semantic retrieval lies outside this intervention.

External validity. The controlled retrieval audit covers all 1,319 GSM8K test questions and three retrievers, while the response bank covers 100 questions and two generators. The naturally written

incorrect-solution audit uses 200 retrieval items and 100 replay items, with independent-model filtering followed by two-annotator human review. LongMemEval-S adds all 500 questions and a separate temporal selection metric; this complements long-term conversational-memory benchmarks such as LoCoMo (Maharana et al., 2024) and incremental memory evaluations such as MemoryAgentBench (Hu et al., 2026) without conflating their different task definitions. The combined design covers three retrievers and two corruption rules. GSM8K answers remain automatically verifiable and open-ended planning, tool use, dialogue, and changing world state may be harder to validate. The BGE-M3 LongMemEval evaluation uses the same encoder across items, with a local implementation used when the embedding service was unavailable.

Statistical validity. For response-bank analyses, two or three seeds provide repeated generation measurements over 100 questions. We cluster by question and treat repeated rows as repeated measurements. The semantic–random interval is $[-3.0, 10.5]$ percentage points. The 1,319-question and 500-question retrieval audits use the retrieval units defined by their benchmarks. The verified and corrupted intervention effects describe the controlled memory setting; natural memory stores require separate validation. Reported p-values are unadjusted and correspond to the displayed paired contrasts; they are not interpreted as a familywise significance claim.

F Design Implications

Evaluate retrieval and utilization separately. Retrieval quality and generator use both determine final task accuracy. Memory benchmarks should report episode selection labels and controlled downstream answers alongside final task accuracy.

Use reliability signals. Dense similarity measures topical match. A memory record can also carry execution results, verifier scores, timestamps, source identity, and contradiction links. Reranking can use these fields explicitly rather than encoding them through embedding distance.

Include matched retrieval controls. Compare semantic selection with random selection at the same number of memories and evaluations, and include an oracle to quantify headroom. In this audit, semantic Top-1 reaches 81%, compared with 93% for no memory and 99% for verified retrieval.

Conflict-aware retrieval. Top-1 selects one episode when the score margin is small. Returning multiple candidates can let a generator cross-check them, as the two-memory conditions illustrate. This design is consistent with conditional retrieval and explicit evidence critique (Asai et al., 2024); prospective episodic-memory tests can select the multi-retrieval threshold on development data.

G Toward Naturalistic Memory Benchmarks

The present construction controls the intervention. A complementary benchmark should begin from actual agent trajectories. Existing long-term memory benchmarks already move in this direction by using multi-session conversations and incremental interactions (Maharana et al., 2024; Wu et al., 2024; Hu et al., 2026). Each stored episode would include the task state, proposed action or solution, environment feedback, execution outcome, and time. Reliability labels should derive from verifiable environment events where possible, with LLM judgments as supplementary evidence. Naturally occurring failures could then be paired with successful episodes matched for task, length, age, and lexical overlap.

Such a benchmark needs three evaluation layers. The *storage* layer asks whether an agent writes failed attempts, corrections, and provenance faithfully. The present study assumes both episodes already exist; writing remains a separate layer. The *retrieval* layer asks whether the system returns relevant and reliable evidence under a fixed budget. Pairwise reliability, recall of verified episodes, corrupted exposure, and abstention should be reported separately. The *utilization* layer intervenes on retrieved context and measures downstream decisions, including whether the model follows an error, resolves a conflict, or requests verification.

Evaluation should also reflect temporal change. A previously correct memory can become outdated after an API, policy, user preference, or world state changes. In that case, historical execution suc-

cess alone is insufficient reliability. Version, timestamp, and dependency metadata become part of retrieval. Conversely, an episode that failed in one state can remain useful as a warning. Binary reliability labels are a starting point for a richer memory ontology.

Our response-bank method can support these richer tests when the candidate set is small: generate the same task under no memory, each singleton, selected pairs, and relevant orderings. For larger stores, complete powerset precomputation is infeasible. A benchmark can sample matched interventions, prioritize high-conflict candidates, and publish the sampling design. Policies must still share generated candidates wherever possible; otherwise retrieval differences become confounded with fresh stochastic evaluations.

Prospective comparisons. A future study should compare dense similarity, lexical retrieval, cross-encoders, provenance-aware rerankers, outcome filters, and explicit verifiers. Each method needs a random equal-budget control and a reliability oracle. Thresholds for multi-retrieval or abstention should be selected on a development split and evaluated once on a held-out test set. Multiple generators can reveal whether better retrieval reduces dependence on model robustness, while multiple embedding families test whether the observed near-chance ranking is architecture-specific.

Metrics. End-to-end accuracy should be accompanied by selected-memory reliability, coverage, number of injected episodes, prompt tokens, generation count, and latency. When a policy abstains or retrieves several candidates, accuracy at fixed cost and risk–coverage curves are more informative than one operating point. Paired transition matrices show which instances benefit or degrade; aggregate averages alone can hide a small subgroup repeatedly exposed to unreliable memories.

H Broader Impact

Unreliable memory retrieval can cause agents to repeat prior mistakes with increased confidence, particularly when the episode is highly relevant and presented as experience. In education, finance, healthcare, or autonomous tool use, provenance-free memory could preserve errors across sessions. Adding verification and uncertainty-aware retrieval may reduce this risk but also adds computation and can encode verifier biases.

Our data contain public mathematical word problems and no personal interaction. The synthetic corruptions are designed for robustness evaluation. Releasing exact prompts and outputs can support auditing. Deployment safety in high-stakes settings requires dedicated evaluation.

I Reproducibility and Evidence

The supplementary materials contain the 100 frozen response-bank items, verified and corrupted episodes, response-bank configurations, generated responses, parsed answers, token counts, and analysis procedure. Each run is checked for the expected number of evaluations, valid parsing, distinct corruption outcomes, consistent retrieval assignments, and faithful policy replay. The supplementary materials also provide the query-ablation summary and question set, together with preprocessing descriptions for the full-split BGE-M3, naturally written incorrect-solution, and LongMemEval evaluations.

Reproduction first constructs the frozen items, generates the five contexts per question for each seed, verifies each run, and then combines policy rows by question. This procedure distinguishes regeneration of stochastic model text from exact replay of the reported policies.