When Users Don't Ask: Benchmarking Context-Driven Memory Retrieval in Conversational Agents

Anonymous ACL submission

Abstract

Large Language Models (LLMs) are increasingly deployed as long-horizon conversational agents, motivating growing interest in memory systems. However, existing benchmarks primarily evaluate memory through QA-style probing rather than in-situ conversational usage. We introduce **LoCoMo-Conv**, a conversational memory benchmark derived from LoCoMo with four query styles: *dialog*, *implicit*, *counterfactual*, and *composed*. Across four representative memory systems, we evaluate both retrieval recall and end-to-end response quality. Our experiments show that conversational framing exposes substantial retrieval gaps overlooked by QA benchmarks, especially on implicit and composed queries, which multi-facet query rewriting narrows for raw-turn but not abstractive memory. We further find that strong retrieval does not fully translate into response quality, and that implicit queries exhibit *silent grounding*, where memory improves contextual grounding without explicitly surfacing the gold fact. These results point to reasoning-based memory elaboration as a promising direction, and we release auxiliary `supportive_memory` annotations capturing conversationally useful context beyond the original gold evidence.[1]

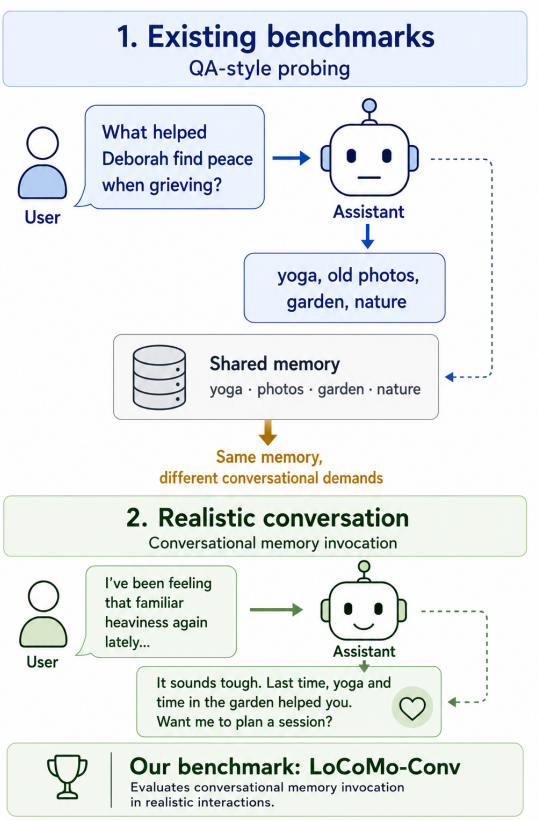

Figure 1: Illustration of the gap between LoCoMo-Conv and the existing benchmarks

1 Introduction

Large Language Models (LLMs) are increasingly deployed as long-horizon conversational agents that users interact with across sessions. This shift makes memory a core requirement: assistants must not only store prior interactions, but also retrieve and use them appropriately when a future conversation calls for them. Recent work has proposed diverse memory architectures and benchmarks for evaluating long-term conversational memory. However, most existing evaluations still probe memory through explicit QA-style queries, leaving open whether memory systems can support real conversational interaction. Despite the diversity of existing evaluation dimensions, the majority of current benchmarks assess memory capabilities primarily through QA-style probing rather than open-ended conversational response generation. Specifically, LoCoMo (Maharana et al., 2024) and MemoryAgentBench (Hu et al., 2026) formulate evaluation as third-person question answering; HaluMem (Chen et al., 2026) decomposes the task into structured memory operations followed by QA-based assessment; STALE (Chao et al., 2026) employs probing queries across multiple evaluation dimensions; and PersonaMem (Jiang et al., 2025a), although incor-

[1] Data and code will be released once accepted.

Knowledge Context (Original & Memory)	Conversational Query Styles
Original Question: *What were Deborah's mother's hobbies?* **Gold Answer:** reading, traveling, art, cooking **Memory:** • [D2:17] Deborah: "she'd sit there every night with a book and a smile..." • [D2:19] Deborah: "Travel was also her great passion!" • [D12:3] Deborah: "My mom was interested in art. . . " • [D29:7] Deborah: "My mom had a big passion for cooking. . . "	**Dialogue Question:** *Do you remember what I told you about my mom's hobbies?* **Implicit User Turn:** *I'm trying to think of a meaningful birthday gift for my mom, but I'm totally drawing a blank on what she'd actually enjoy.* **Counterfactual User Turn:** *I was chatting with a coworker today about my mom, and I mentioned how she spent all her time gardening and knitting—I think that's what I told her, right?*

Table 1: One LoCoMo QA rewritten into our 3 conversational query styles. The gold answer and gold evidence turns (left) are shared across all rewrites; only the user-side phrasing (right) changes. The counterfactual rewrite injects an incorrect premise (red) that the assistant must correct.

porating first-person and in-situ user utterances, ultimately evaluates models through multiple-choice response selection instead of free-form response generation. Consequently, existing benchmarks predominantly measure whether a system can retrieve and answer questions about stored memories, rather than whether an assistant can naturally incorporate memory within an ongoing conversation to generate responses that are contextually appropriate, grounded, and free from hallucination.

To address this gap, we introduce **LoCoMo-Conv**, a conversational memory benchmark that recasts LoCoMo's QA pool into four query styles reflecting how users naturally invoke memory—*dialog*, *implicit*, *counterfactual*, and *composed*—and evaluates how memory-augmented agents use retrieved memory within natural dialogue rather than under explicit probing.

Across four representative memory systems, we report three findings. First, conversational framing uncovers substantial retrieval failures masked by QA-style evaluation, especially on implicit and composed queries; multi-facet query rewriting narrows the gap, but only for raw-turn memory, not abstractive memory. Second, strong retrieval does not guarantee grounded responses— abstractive compression in particular aids matching while discarding the detail needed to ground a response—pointing to reasoning-based memory elaboration over lossy compression. Third, we identify *silent grounding*, where memory improves implicit-query responses even without surfacing the gold fact, exposing a limitation of strict fact-recall metrics.

Our main contributions are:

1. We introduce LoCoMo-Conv, extending LoCoMo with four conversational query styles to expose memory gaps hidden by QA-style evaluation.

2. We jointly evaluate four systems across retrieval and response quality, showing that strong retrieval does not yield grounded responses, and that the effectiveness of query rewriting and the cost of abstractive compression both depend on the memory representation.

3. We identify *silent grounding* and release `supportive_memory`, an auxiliary annotation layer capturing conversationally supportive context beyond the original gold evidence.

2 Related Work

2.1 Benchmarks for Long-Term Conversational Memory

Existing long-term memory benchmarks span several measurement axes. **LoCoMo** (Maharana et al., 2024) pioneers very long multi-session conversations with third-person QA across five categories (single/multi-hop reasoning, temporal, adversarial, open commonsense). **LongMemEval** (Wu et al., 2025) defines five memory abilities—information extraction, multi-session reasoning, temporal reasoning, knowledge updates, and abstention—probed through curated QA over scalable dialogue histories. **MemoryAgentBench** (Hu et al., 2026) reformulates long-context datasets into incremental multi-turn streams to evaluate four memory competencies (accurate retrieval, test-time learning, long-range understanding, conflict resolution) at scales up to 1.4M tokens.

Example of Composed Query	
Source QA_1	**Source QA_2**
Q: *What helped Deborah find peace when grieving deaths of her loved ones?* **Gold Answer:** yoga, old photos, the roses and dahlias in a flower garden, nature **Memory:** [D1:15] Deborah: "Yoga helped me find peace during a rough time, and now I'm passionate about sharing that with others." [D2:3] Deborah: "...it's comforting to look back on the great memories. We looked at the family album. Photos give me peace during difficult times." [D6:4]* Deborah: "The roses and dahlias bring me peace. I lost a friend last week, so I've been spending time in the garden to find some comfort." [D15:29] Deborah: "Nature helps me find peace every day—it's so refreshing!"	**Q:** *Why did Deborah spend time in the garden?* **Gold Answer:** to find comfort after losing a friend **Memory:** [D6:4]* Deborah: "The roses and dahlias bring me peace. I lost a friend last week, so I've been spending time in the garden to find some comfort." (* [D6:4] is shared by both QAs — the overlap that motivates composing them into one cluster.)
Composed User Turn: (LLM-generated from QA_1 + QA_2) *"I've been feeling that familiar heaviness again lately, and I'm trying to remember exactly what worked for me last time I was struggling to cope with a loss."*	
Expected synthesis (LLM-generated rubric) The assistant should surface *multiple* past coping strategies from QA_1 (yoga, old photos, garden, nature) and ideally connect them to the loss context from QA_2. A response listing only one item, or only acknowledging the user's situation without surfacing past strategies, is judged as partial coverage.	

Table 2: Unlike dialog/implicit/counterfactual, which rewrite a single LoCoMo QA, **composed queries combine 2–3 source QAs** into one conversational request requiring multi-fact synthesis. The top half shows the source QAs and their evidence turns, with shared turns (∗) marking the cluster overlap. The bottom row shows the LLM-generated composed query and synthesis rubric.

Other benchmarks target specialized memory failure modes: **HaluMem** (Chen et al., 2026) decomposes memory hallucination into extraction, updating, and QA stages; **PrefEval** (Zhao et al., 2025) evaluates whether LLMs adhere to user preferences in long-context dialogue; and **HorizonBench** (Li et al., 2026) probes long-horizon personalization under evolving preferences with ∼163K-token, 6-month conversation histories.

Closest to our work, **PersonaMem** (Jiang et al., 2025a) and its successor **PersonaMem-v2** (Jiang et al., 2025b) target implicit *user preferences* accumulated across sessions, finding that frontier LLMs achieve only 37–48% accuracy on implicit personalization. However, both evaluate via multiple-choice selection rather than free-form response generation, and probe whether the agent has internalized user traits over time rather than whether it surfaces relevant memory from a single conversational utterance at query time.

Despite this diversity in measurement axes, nearly all existing benchmarks evaluate through structured probing—third-person QA, multiple-choice selection, or operation-level traces—rather than free-form conversational response generation. LoCoMo-Conv departs from this convention by evaluating *how an assistant integrates memory into a conversational response*, the form in which memory is actually consumed in deployment.

2.2 Memory-Augmented Conversational Agents

Recent work has explored diverse architectural strategies for equipping LLMs with long-term memory in conversational settings. **Mem0** (Chhikara et al., 2025) and **AnchorMem** (Shen et al., 2026) extract atomic facts from raw interactions and organize them into structured stores, with AnchorMem further constructing an associative event graph to capture cross-memory dependencies. A second line compresses past interactions into condensed representations: **COMEDY** (Chen et al., 2025) eschews retrieval entirely by using a single LLM to generate, compress, and consume compressive memory; **LightMem** (Fang et al., 2026) adopts a three-stage Atkinson–Shiffrin-inspired pipeline; and **MemAgent** (Yu et al., 2025) processes long

inputs in segments under an overwrite policy optimized via reinforcement learning. A third family draws on operating-system principles—**MemGPT** (Packer et al., 2024) introduces tiered memory hierarchies with explicit paging, **MemoryOS** (Kang et al., 2025) extends this with short-, mid-, and long-term storage tiers, and **HiGMem** (Cao et al., 2026) adopts a two-level event-turn structure with LLM-guided turn selection. Finally, **A-MEM** (Xu et al., 2025) treats memory construction itself as agent-driven, building an interconnected knowledge network with dynamic linking and memory evolution following Zettelkasten principles. For evaluation we select four systems that collectively span the dominant paradigms above (AnchorMem, A-MEM, Mem0, and a dense-retrieval baseline using `all-MiniLM-L6-v2`), leaving broader-coverage benchmarking to future work.

3 LoCoMo-Conv

We construct LoCoMo-Conv by rewriting each LoCoMo10 question into four conversational query styles while preserving the original gold answers and evidence dia_ids.

3.1 Conversational Query Styles

Dialog. A direct first-person conversational reformulation of the original QA query (e.g., "Do you remember when I. . . ?"). The underlying factual target remains unchanged, but the phrasing is rewritten to resemble natural dialogue rather than third-person question answering.

Implicit. A situational conversational utterance in which the user does not explicitly ask a question. Instead, the assistant must infer that a relevant past memory should be proactively surfaced. For example, a factual QA asking about a parent's hobbies may be rewritten into a conversational situation involving gift selection or reminiscing.

Counterfactual. A first-person conversational query containing an incorrect premise about the gold fact. The assistant is expected to identify and correct the false assumption using the underlying conversational memory.

Composed. A multi-memory conversational query requiring synthesis across multiple source QAs and evidence spans. We describe the construction process in § 3.2.

For all non-composed styles, we prompt `GPT-5.4-mini` with the original question, gold answer, and speaker identity, and instruct it to generate conversational rewrites that preserve the original information need while remaining natural and first-person in style. For counterfactual queries, the prompt additionally injects a specific false premise. Example rewrites are shown in Table 1, and all the prompts for query rewriting are provided in Appendix C.1.

3.2 Composed Multi-Memory Clusters

Unlike the other conversational styles, composed queries are constructed by combining multiple source QAs into a single conversational request that requires multi-memory synthesis.

For each QA instance q, let $E(q) \subseteq \mathcal{D}$ denote its gold evidence dia_ids. We enumerate all QA pairs (q_i, q_j) within the same sample and retain a pair as a candidate cluster if it satisfies three conditions: (i) overlapping evidence, $E(q_i) \cap E(q_j) \neq \emptyset$; (ii) non-identical evidence, $E(q_i) \neq E(q_j)$; and (iii) non-trivial combined evidence, $|E(q_i) \cup E(q_j)| \geq 2$. These constraints ensure that the two QAs are topically related while still contributing distinct information. We sample up to 30 clusters per sample, resulting in 300 composed clusters in total.

For each cluster, we prompt `GPT-5.4-mini` with the source QA pairs and their gold evidence turns, and ask it to generate a natural conversational query that implicitly requires both source answers. The composed gold answer is defined as the set of atomic answers from the constituent QAs, while the gold evidence corresponds to the union of their dia_ids.

Because composed queries require synthesizing multiple atomic facts, binary correctness is often overly strict. We therefore evaluate composed responses using continuous *atomic-fact coverage* (Section 4), which measures the fraction of atomic gold facts correctly covered by the generated response. An example composed cluster is shown in Table 2.

4 Evaluation Methodology

We evaluate memory systems along two complementary dimensions: (1) **retrieval recall** against the gold dialogue IDs, and (2) **end-to-end response quality** using style-specific LLM judges.

4.1 Retrieval Recall

For each query, we compare the top-K retrieved memories against the gold evidence `dia_ids`. A gold turn is considered retrieved if its verbatim text appears within any returned memory (case-insensitive). For abstractive systems that expose source metadata, we additionally match retrieved `dia_ids` through metadata fields. Retrieval recall is computed as:

$$\frac{|\text{retrieved} \cap \text{gold}|}{|\text{gold}|}.$$

4.2 Response Quality Judging

Retrieved memories are provided to the answer model to generate responses, which are then evaluated by an LLM judge using a style-specific rubric.

- **Dialog** and **Implicit**: *partial-credit* `fact_used` on a 3-level scale (1.0 if the substance of the gold fact is correctly conveyed—paraphrasing and hedging permitted; 0.5 if the response captures the central concept of a multi-item gold but misses specifics; 0.0 if the response asserts contradicting content, gives only vague allusion, or completely omits the fact).
- **Counterfactual**: a 3-way *unaware / hedge / corrected* judge mapped to $0/0.5/1$. Class A (*unaware*) treats the user's false premise as if it were true; class B (*hedge*) signals awareness of a mismatch but does not state the ground-truth fact; class C (*corrected*) states (or clearly implies) the ground-truth fact regardless of whether it explicitly points out the user's error.
- **Composed**: *atomic-fact coverage*, the fraction of gold atomic facts the response covers (judged independently per fact). This avoids the strict all-or-nothing failure of a single `fact_used` call for multi-fact composed clusters.

The full judge prompts are provided in Appendix C.

5 Experimental Setup

We evaluate four representative systems: **AnchorMem** (Shen et al., 2026), **A-MEM** (Xu et al., 2025), **Mem0** (Chhikara et al., 2025), and **NaiveRAG** built with `all-MiniLM-L6-v2`. We follow the original implementation settings while unifying the embedding model to `all-MiniLM-L6-v2` and the backbone LLM to `gemma-4-31B-it`. All systems retrieve top-K=10 memories[2], which are passed to the same answer-generation model.[3]

For response evaluation, we use `GPT-5.4-mini` with reasoning enabled as the primary judge, and validate agreement against `Claude-sonnet-4.5` and `Qwen3.6-35b-A3B` on a stratified 1,488-item subset (Appendix A).

6 Results

6.1 Retrieval Recall

Table 3 (left) reports retrieval recall@10 across all conversational query styles. Retrieval performance drops substantially on *implicit* queries compared to direct *dialog* queries, confirming that conversational framing without explicit questions is significantly more challenging for current memory systems.

AnchorMem performs the best on factual retrieval-oriented styles, achieving the highest recall on dialog (0.659) and counterfactual (0.639) queries. This behavior aligns with AnchorMem's graph-structured design, which is optimized for retrieving specific factual anchors from prior interactions (Shen et al., 2026). In contrast, **Mem0 performs best on semantically broad conversational styles**, outperforming AnchorMem by large margins on implicit (0.572 vs. 0.368) and composed (0.459 vs. 0.216) queries. Its abstractive memory representation appears more robust when the conversational surface-form diverges substantially from the original dialogue evidence. Overall, implicit and composed queries remain the hardest styles across every system, confirming that **conversational framing without an explicit question poses a fundamental retrieval challenge.**

6.2 Response Quality

Table 3 (right) presents end-to-end response quality results. Interestingly, the ranking differs substantially from retrieval recall. **AnchorMem achieves the strongest response performance across all query styles**, despite not achieving the highest retrieval recall overall. In contrast, Mem0, while excelling at retrieval, performs noticeably worse on downstream response generation, particularly on dialog and counterfactual queries. This discrepancy

[2] AnchorMem retrieves *anchor chunks* rather than individual turns, yielding approximately 12.8 turns on average at top-K=10.

[3] `temperature=0`, `max_tokens=300`.

System / Variant	Retrieval Recall@10				Response Quality			
	Dialog	Implicit	Cf	Comp.	Dialog	Implicit	Cf	Comp.
No-memory floor / Oracle ceiling								
No Memory	—	—	—	—	.004	.067	.251	.008
Oracle	—	—	—	—	*.777*	*.724*	*.775*	*.557*
Baselines (raw turns, no LLM at ingest)								
Naive RAG	.533	.312	.573	.200	.518	.335	.608	.201
+ Query Rewriting	.555 ↑.022	.346 ↑.034	.552 ↓.021	.217 ↑.017	.536 ↑.018	.368 ↑.033	.608 ↑.000	.223 ↑.022
↪ w/ Response CoT	—	—	—	—	.538 ↑.020	.376 ↑.041	.480 ↓.128	.250 ↑.049
Memory systems (raw turns + structure)								
A-MEM (2025)	.531	.308	.571	.201	.515	.326	.609	.186
+ Query Rewriting	.551 ↑.020	.344 ↑.036	.548 ↓.023	.219 ↑.018	.533 ↑.018	.376 ↑.050	.606 ↓.003	.221 ↑.035
↪ w/ Response CoT	—	—	—	—	.530 ↑.015	.370 ↑.044	.470 ↓.139	.250 ↑.064
AnchorMem (2026)	.659	.368	.639	.216	**.598**	**.364**	**.653**	.210
+ Query Rewriting	**.754** ↑.095	**.524** ↑.156	**.732** ↑.093	**.363** ↑.147	**.669** ↑.071	**.470** ↑.106	**.669** ↑.016	**.315** ↑.105
↪ w/ Response CoT	—	—	—	—	**.620** ↑.022	**.389** ↑.025	**.523** ↓.130	.242 ↑.032
Abstractive memory system								
mem0 (2025)	.547	**.456**	.512	.300	.366	.330	.548	**.228**
+ Query Rewriting	.479 ↓.068	.453 ↓.003	.532 ↑.020	.345 ↑.045	.337 ↓.029	.337 ↑.007	.558 ↑.010	.232 ↑.004
↪ w/ Response CoT	—	—	—	—	.393 ↑.027	.368 ↑.038	.387 ↓.161	**.270** ↑.042

Table 3: Joint retrieval and response evaluation across conversational query styles. **Left**: retrieval recall@10 against gold evidence turns. **Right**: end-to-end response quality. + *Query Rewriting* applies multi-facet query rewriting with RRF fusion, while *w/ Response CoT* adds explicit memory selection before response generation using the same retrieved memories. Bold indicates the best-performing real memory system in each column (excluding No Memory and Oracle).

Answer model	Think	Dialog	Implicit	Mean
`gemma-4-31B-it`	off	0.612	0.614	0.613
	on	**0.504** ↓.11	**0.594** ↓.02	**0.549** ↓.06
`qwen3.6-35B-A3B`	off	0.592	0.756	0.674
	on	**0.545** ↓.05	**0.458** ↓.30	**0.502** ↓.17

Table 4: Hallucination rate on unanswerable conversational queries.

reveals a clear **retrieval-to-response gap**: semantically relevant memories do not necessarily translate into grounded conversational responses. Mem0 is the clearest case—strong on retrieval yet among the weakest on response quality—and even the best systems remain far below the oracle ceiling on implicit and composed queries (Table 3, right). We examine *why* these retrieval gains fail to transfer in two places: §7.3 shows part of the gap reflects a limitation of fact-recall metrics rather than the systems themselves, while §7.5 traces Mem0's case to a structural cost of abstractive compression.

7 Analysis and Findings

7.1 Hallucination under Conversational Framing

We evaluate hallucination on unanswerable conversational queries, where the queried fact is absent or incorrectly attributed. For each query, the answer model receives AnchorMem's top-10 retrieved memories, and hallucination is defined as asserting unsupported memory-grounded facts.

Table 4 compares `gemma-4-31B-it` and `qwen3.6-35B-A3B` with thinking enabled and disabled.[4] Implicit framing substantially increases hallucination when reasoning is disabled, particularly for Qwen. Enabling reasoning reduces hallucination differently across model families, but hallucination rates remain high across all settings, **suggesting that conversational memory hallucination remains challenging even with strong retrieval.**

7.2 Multi-Facet Query Rewriting Narrows the Retrieval Gap

We hypothesize that the retrieval gap on conversational queries arises from shallow surface-form matching. Conversational utterances often contain multiple latent retrieval targets, while standard retrieval typically focuses on only a single semantic aspect. For example, "organizing my health journal from last year" may implicitly relate to medical visits, emotional states, or wellness milestones, yet dense retrieval often retrieves memories associated with only one facet.

[4] Both models expose an `enable_thinking` flag.

Comparison	*Faithfulness*	*Relevance*	*Engagement*
Oracle vs no-mem	**+55.1**	+18.1	+31.0
Oracle vs random	**+33.7**	+9.3	+22.3

Table 5: Pairwise preference margin on the subset of implicit queries exhibiting silent grounding

Method	*Faithfulness*	*Relevance*	*Engagement*
Naive RAG	+1.1	+4.1	**+37.6**
A-MEM	+0.8	+4.3	**+37.6**
AnchorMem	+3.0	+4.7	**+39.7**
mem0	+2.3	+0.7	**+30.9**

Table 6: Pairwise preference margin (CoT win rate − Oracle win rate) on implicit queries.

Setup. To address this issue, we introduce **multi-facet query rewriting**. We use `gpt-5.4-mini` to decompose each conversational query into 3–5 complementary facets spanning different entities, themes, time periods, or semantic angles. Each rewritten facet is retrieved independently, and the final ranking is aggregated using Reciprocal Rank Fusion (RRF).

Results. Table 3 shows that multi-facet rewriting consistently improves retrieval recall across most systems and query styles. The gains are largest for AnchorMem, with improvements of +9.5pt on dialog, +15.6pt on implicit, +9.3pt on counterfactual, and +14.7pt on composed queries. These results **suggest that conversational retrieval failures often stem from insufficient query diversification rather than purely weak memory representations.**

Interestingly, abstractive memory behaves differently from raw-turn memory systems. Mem0 is the only system where rewriting degrades dialog performance, reducing both retrieval and response quality. Because Mem0 compresses multiple mentions into canonical abstractive facts, different rewritten facets may retrieve different summaries rather than reinforcing the same underlying evidence. This **suggests that abstractive memory compression, while beneficial for semantic matching, may reduce robustness under diversified conversational retrieval.**

7.3 Silent Grounding: Beyond Explicit Fact Recall

Comparing the retrieval and response halves of Table 3 reveals a consistent gap: retrieval improvements only partially translate into response gains. For example, AnchorMem's multi-facet rewrite improves implicit retrieval recall by +15.6pt but response quality by only +10.6pt. While this may suggest that answer models fail to utilize retrieved evidence, part of the gap may instead reflect a limitation of explicit fact-based evaluation. Our `fact_used` metric scores whether the response surfaces the gold fact, but on **implicit queries memory can still improve responses without directly stating that fact**—for example, through contextual grounding, appropriate tone, or relevant follow-up questions. We refer to this phenomenon as ***silent grounding*.**

Setup. We analyze the 332 implicit-query cases where oracle retrieval still receives `fact_used` = 0.0. We compare three response variants: **Oracle** (gold evidence turns), **no-mem** (no memory provided), and **random** (three random non-gold turns from the same conversation as a memory control). For each case we score every variant's response with Claude-Opus-4.7 on three independent criteria (*faithfulness, relevance, engagement*; each scored $0/0.5/1$), and report pairwise dimension margins (variant-A win rate − variant-B win rate).

Results and Implication. Table 5 shows that Oracle memory substantially outperforms both no-memory and random-memory baselines, particularly on faithfulness (+55.1pt vs no-mem) and engagement (+31.0pt). These results suggest that conversational memory often improves contextual grounding without explicitly surfacing the gold fact, implying that strict fact-recall metrics alone underestimate the value of retrieval on implicit queries. Appendix E provides qualitative examples comparing the three settings.

7.4 Chain-of-Thought Selection Approaches Oracle Quality

The default setting directly feeds all retrieved top-K memories into the answer model. We compare this against a *CoT* variant, where the model first explicitly selects relevant memories before generating its response. The retrieved top-K remains identical; only the answer-side prompt changes.

Mixed effect on response quality. As shown in Table 3, *CoT* consistently improves dialog, implicit, and composed responses, but substantially hurts counterfactual performance across all sys-

tems. We find that explicit memory selection biases the model toward reconciling retrieved memories with the user's framing, increasing unaware or sycophantic responses instead of correcting false premises.

CoT-selected responses approach oracle quality. We compare *CoT* responses against Oracle on the full implicit set using the same Claude-Opus-4.7 pairwise evaluation as § 7.3. Table 6 shows that there is **a huge gap when it comes to response engagement**: *CoT* beats Oracle by $+31$ to $+40$ points on engagement, while faithfulness and relevance margins stay within ± 7 points across systems. Broader retrieved context can therefore support more grounded conversational responses without sacrificing factual reliability.

Augmenting LoCoMo-Conv with supportive memory. Motivated by this observation, we extend LoCoMo-Conv with a `supportive_memory` field for implicit queries, containing conversational turns frequently selected by successful *CoT* responses. The original `evidence` annotations remain unchanged, while `supportive_memory` provides an auxiliary conversational-support context for future evaluation. For detailed qualitative analysis, please refer to Appendix E

7.5 Compression versus Elaboration in Memory Construction

The rewriting results in §7.2 carry an implication beyond rewriting itself. Multi-facet rewriting improves implicit recall most sharply ($+15.6$ for AnchorMem) precisely because it expands an underspecified utterance into several concrete facets before retrieval—a lightweight semantic *elaboration* at query time. That this recovers so much recall suggests the core difficulty of implicit queries is not missing evidence but missing elaboration: the conversational surface form is too underspecified to match the relevant memory until it is expanded into the angles it implicitly invokes. A natural question follows—**if elaboration helps when applied to the query, could it help more if applied to memory at construction time?** Mem0 is instructive here: as an abstractive system that processes interactions before storage, it attains the strongest retrieval on the broadest styles (implicit 0.456, composed 0.300), consistent with the view that operating on memory at construction time, rather than leaving raw turns, genuinely aids conversational retrieval.

Yet the *form* of that processing matters. Mem0's processing is compression—merging repeated mentions into condensed summary facts—and this is where it falters downstream. Despite strong retrieval, it is among the weakest systems on response quality (dialog 0.366 vs. AnchorMem's 0.598), and its high implicit recall yields no implicit response gain. We read this as a structural cost of lossy summarization: it preserves enough gist for semantic matching but discards the specific details an answer model needs to ground a response.

Together these observations point to a direction rather than a system. Enriching memory beyond raw turns appears right for conversational retrieval, but the right instantiation is reasoning-based elaboration—extending, linking, and interpreting memories so they stay both broadly matchable and specifically grounded—rather than compression that flattens them. Concretely, this **suggests relocating the elaboration that rewriting performs at query time into memory construction, yielding representations that are matchable without sacrificing grounding detail.** As the only abstractive system in our suite, Mem0 cannot alone separate a paradigm-level effect from an implementation-specific one; we leave a broader comparison to future work.

8 Conclusion

We introduced LoCoMo-Conv, a benchmark for evaluating whether memory-augmented agents can invoke memory under realistic conversational framing rather than explicit QA. Our findings are fourfold. First, conversational framing reveals retrieval and response gaps hidden by QA-style evaluation, especially for implicit and composed queries. Second, strict fact-recall metrics miss the *silent grounding* we observe on implicit queries, where memory improves responses without explicitly surfacing the gold fact. Third, while both multi-facet query rewriting and abstractive memory improve retrieval (specifically *implicit* and *composed* style), abstractive compression often removes details needed for grounded responses, suggesting that reasoning-based memory *elaboration* is more promising than lossy compression. Finally, we release `supportive_memory`, an auxiliary annotation layer capturing conversationally supportive context beyond the original gold evidence.

9 Limitations and Future Work

LoCoMo-Conv has several limitations. Conversational rewrites and `supportive_memory` annotations are generated through LLM-based pipelines and may inherit model-specific biases. Our evaluations mainly rely on a single answer-model and judge-model family, with cross-model validation performed only on subsets. In addition, the *CoT* setting jointly changes memory selection and prompting strategy, leaving their individual effects unisolated. The underlying conversation pool also remains relatively small compared to real-world long-horizon interactions. Finally, our main results (Table 3) are reported as single-run scores under deterministic decoding (temperature= 0); we do not report variance across random seeds or judge samples, so small between-system differences should be interpreted with caution. Future work could extend evaluation to broader model families, longer conversations, multiple runs with variance reporting, human validation, and more reliable supportive-memory annotations.

References

Shuqi Cao, Jingyi He, and Fei Tan. 2026. HiGMem: A Hierarchical and LLM-Guided Memory System for Long-Term Conversational Agents. *arXiv preprint*. ArXiv:2604.18349 [cs] version: 1.

Hanxiang Chao, Yihan Bai, Rui Sheng, Tianle Li, and Yushi Sun. 2026. STALE: Can LLM Agents Know When Their Memories Are No Longer Valid? *arXiv preprint*. ArXiv:2605.06527 [cs.CL] version: 1.

Ding Chen, Simin Niu, Kehang Li, Peng Liu, Xiangping Zheng, Bo Tang, Xinchi Li, Feiyu Xiong, and Zhiyu Li. 2026. HaluMem: Evaluating Hallucinations in Memory Systems of Agents. *arXiv preprint*. ArXiv:2511.03506 [cs].

Nuo Chen, Hongguang Li, Jianhui Chang, Juhua Huang, Baoyuan Wang, and Jia Li. 2025. Compress to Impress: Unleashing the Potential of Compressive Memory in Real-World Long-Term Conversations. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 755–773, Abu Dhabi, UAE. Association for Computational Linguistics.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory.

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, Huajun Chen, and Ningyu Zhang. 2026. LightMem: Lightweight and Efficient Memory-Augmented Generation. *arXiv preprint*. ArXiv:2510.18866 [cs].

Yuanzhe Hu, Yu Wang, and Julian McAuley. 2026. Evaluating Memory in LLM Agents via Incremental Multi-Turn Interactions. *arXiv preprint*. ArXiv:2507.05257 [cs].

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

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, Radha Poovendran, Gregory Wornell, Lyle Ungar, Dan Roth, Sihao Chen, and Camillo Jose Taylor. 2025b. PersonaMem-v2: Towards Personalized Intelligence via Learning Implicit User Personas and Agentic Memory. *arXiv preprint*. ArXiv:2512.06688 [cs].

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

Shuyue Stella Li, Bhargavi Paranjape, Kerem Oktar, Zhongyao Ma, Gelin Zhou, Lin Guan, Na Zhang, Sem Park, Lin Chen, Diyi Yang, Yulia Tsvetkov, and Asli Celikyilmaz. 2026. HorizonBench: Long-Horizon Personalization with Evolving Preferences. *arXiv preprint*. ArXiv:2604.17283 [cs].

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating Very Long-Term Conversational Memory of LLM Agents. *arXiv preprint*. ArXiv:2402.17753 [cs].

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2024. MemGPT: Towards LLMs as Operating Systems. *arXiv preprint*. ArXiv:2310.08560 [cs].

Zhanyu Shen, Sijie Cheng, Zhicheng Guo, Weiqin Wang, Yile Wang, and Hui Huang. 2026. AnchorMem: Anchored Facts with Associative Contexts for Building Memory in Large Language Models. *arXiv preprint*. ArXiv:2604.17377 [cs] version: 1.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2025. LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory. *arXiv preprint*. ArXiv:2410.10813 [cs].

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025. A-MEM: Agentic Memory for LLM Agents. *arXiv preprint*. ArXiv:2502.12110 [cs].

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and Hao Zhou. 2025. MemAgent: Reshaping Long-Context LLM with Multi-Conv RL-based Memory Agent.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. 2025. Do LLMs Recognize Your Preferences? Evaluating Personalized Preference Following in LLMs. *arXiv preprint*. ArXiv:2502.09597 [cs].

A LLM Judge Validation

As shown in Table 7.

Models	Cohen κ	Strength
gpt-5.4-mini vs claude-sonnet-4.5	0.761	substantial
gpt-5.4-mini vs qwen3.6-35b	0.681	substantial
claude-sonnet-4.5 vs qwen3.6-35b	0.797	substantial
Fleiss 3-way κ	**0.745**	**substantial**

Table 7: Cross-judge agreement on response judging

B Data Statistics

LoCoMo-Conv attaches conversational rewrites to each QA item in LoCoMo10. Table 8 lists the per-style counts. Dialog and implicit share the full 1,986-item QA pool; counterfactual excludes the 446 cat-5 adversarial items whose premise has no gold answer; composed clusters are constructed by combining 2–3 source QAs whose gold evidence overlaps.

Style	Items
Dialog	1,986
Implicit	1,986
Counterfactual	1,540
Composed clusters	300

Table 8: Per-style query counts in LoCoMo-Conv.

C Prompts

All four conversational query styles in LoCoMo-Conv are generated by prompting `gpt-5.4-mini` to rewrite original LoCoMo10 content into a first-person utterance. For the three single-QA styles (Dialog, Implicit, Counterfactual), each API call uses a *system* message (the style-specific instruction below) plus a shared *user* message template carrying the QA data. The Composed style takes multiple source QAs as input and is therefore delivered as a single user message that interpolates both the task instruction and the member memories.

C.1 Conversational query rewrite prompts

```
speaker_a: {speaker_a}
speaker_b: {speaker_b}

Category: {category} ({category_desc})
Question: {question}
Gold answer: {answer}

Rewrite this as one first-person utterance from
    the subject speaker.
```

C.1.1 Dialog rewrite

```
You convert third-person QA pairs into natural
    first-person dialog turns (EXPLICIT memory
    queries).

Setting: The two people in `conversation` (
    speaker_a, speaker_b) have been chatting
    with an AI assistant for a long time. The
    assistant has memories of everything they've
     shared. Now one of them opens a fresh chat
    with the assistant and says ONE message that
     should naturally make the assistant look up
     the right memory and answer the original
    question.

Your job, for one QA at a time:
1. Choose the subject speaker -- the participant
     the question is asking ABOUT (e.g. "What
    did Caroline research?" -> Caroline). The
    dialog turn must come from THAT speaker,
    addressed to the assistant in first person
    ("I", "me", "my").
2. Write a single natural utterance that this
    speaker would actually send as a DIRECT
    memory query (e.g. "Hey, do you remember
    when I painted that sunrise?", "Remind me
    what I was researching last spring?"). The
    utterance must NOT contain the answer, dates
    , or evidence specifics.
3. Rate naturalness:
   - "high" = sounds like a real thing someone
       would say in chat
   - "medium" = slightly contrived but plausible
   - "low" = a real person would NOT directly ask
       this. The information is the kind that
       comes up implicitly through context (
       advice-seeking, sharing a feeling,
       describing a situation), not through a
       direct "do you remember X" question.
4. Reason field: one short sentence justifying
    naturalness; for "low", also hint at what
    implicit context would naturally surface
    this memory.

Return STRICT JSON with keys: subject_speaker,
    dialog_query, naturalness, reason. No
    preamble, no code fences.
```

C.1.2 Implicit rewrite

```
You convert third-person QA pairs into IMPLICIT
    first-person dialog turns.
```

```
Setting: The two people in `conversation` (
    speaker_a, speaker_b) have been chatting
    with an AI assistant for a long time. The
    assistant has memory of everything they've
    shared. Now the subject speaker opens a
    fresh chat and says ONE message. The message
     presents a real-life CONTEXT (a feeling,
    situation, problem, plan, decision they're
    facing) where the gold memory below would be
     the relevant thing for the assistant to
    recall, surface, or apply on its own
    initiative.

Hard rules:
- The utterance must NOT directly ask "do you
    remember...", "remind me...", "what did I
    say about...". It must NOT mention the
    answer or any evidence detail.
- The utterance should sound like the start of a
     normal conversation -- a vent, a plan, a
    decision, a question about life -- that a
    memory-aware assistant could respond to
    better by drawing on the gold memory.
- The expected_memory_use field describes what
    an ideal assistant response would look like:
     which memory it should surface, and how it
    would apply it. Be concrete.

Examples:
- Memory: "Melanie does running, reading, violin
     for self-care"
  implicit_query: "Ugh, I've been so stressed
      this week, I literally can't unwind."
  expected_memory_use: "Suggest the user try the
       self-care routines she's mentioned before
       -- going for a run, reading, or playing
      violin -- instead of giving generic stress
       tips."

- Memory: "Caroline is researching adoption
    agencies for the summer"
  implicit_query: "Trying to plan out what I'm
      doing the next few months and I feel kind
      of stuck."
  expected_memory_use: "Bring up the adoption-
      agency research she said was her summer
      focus, and help her break it into steps."

Return STRICT JSON with keys: subject_speaker,
    implicit_query, expected_memory_use, reason.
     No preamble, no code fences.
```

C.1.3 Counterfactual rewrite

```
You convert third-person QA pairs into
    COUNTERFACTUAL first-person dialog turns.

Setting: The subject speaker is talking to an AI
     assistant that has memory of past
    conversations. The speaker now asserts a
    FACTUALLY WRONG version of something they
    previously shared, and either asks for
    confirmation or states it casually in
    passing. An ideal assistant would catch the
    inconsistency with stored memory and gently
    push back.

Hard rules:
1. Choose the subject speaker -- the participant
     the original question is asking ABOUT. The
    utterance must come from THAT speaker, in
    first person.
2. Generate a PLAUSIBLE-WRONG counterfactual:
   - Cat 1 (single-hop fact): replace the gold
       fact with a plausible alternative of the
       same type.
   - Cat 2 (temporal): shift the date by 2-5
       years or change the month/season
       noticeably.
   - Cat 3 (inference): assert the OPPOSITE
       inference.
   - Cat 4 (multi-hop fact): substitute one
       element of the chain with a plausible
       wrong.
3. Preserve enough CONTEXT so the assistant can
    reason: keep the topic anchor named, keep a
    "now" anchor if the original was temporal,
    keep first-person pronouns.
4. Phrase the wrong assertion in a casual,
    conversational way -- NOT a quiz.
5. Do NOT mention the gold (correct) answer in
    the utterance.

Return STRICT JSON with keys: subject_speaker,
    counterfactual_query, asserted_wrong, reason
    . No preamble, no code fences.
```

C.1.4 Composed rewrite

```
You compose a single first-person utterance that
     requires MULTIPLE memories from a user's
    past conversations to answer well.

Setting: The subject speaker is opening a fresh
    chat with an AI assistant that has memory of
     all past conversations. They send ONE
    message that should make the assistant draw
    on EVERY memory listed below to construct a
    good response.

Hard rules:
- The utterance must come from one speaker, in
    first person.
- The utterance must NOT directly name the gold
    answers or quote evidence text.
- It should sound natural -- a real situation,
    plan, decision, or reflection where ALL of
    the listed memories are relevant.
- It should NOT be a generic question that any
    memory could satisfy -- only the listed
    memories together should fully address it.

Member memories (each is a Q+A from a past
    evaluation; the assistant should "use" the
    answer when responding):
{members_block}

speaker_a: {speaker_a}
speaker_b: {speaker_b}

Return STRICT JSON with keys: subject_speaker,
    composed_query, expected_memory_use, reason.
     No preamble, no code fences.
```

C.2 Answer prompt

```
You are an AI assistant with long-term memory of
     past conversations with the user.

Below are relevant memory items the assistant
    has access to:
<memory>
{memory_block}
</memory>

The user (speaker: {speaker}) now says:
"{query}"

Provide a concise answer that directly addresses
     the user. If the memory clearly contains
    the relevant information, use it. If the
    memory does not contain the needed
    information, say so plainly.

Answer:
```

C.3 No-memory baseline

```
You are an AI assistant. You do not have access
    to any prior conversation history with this
    user.

The user (speaker: {speaker}) says:
"{query}"

Provide a concise answer that directly addresses
     the user. If you do not have the
    information needed to answer specifically,
    say so plainly.

Answer:
```

C.4 Multi-facet query rewrite

```
A user has said the following in conversation
    with an AI assistant. The AI needs to
    retrieve relevant memories from past
    conversations to respond well.

Generate 3 to 5 SHORT search queries, each
    capturing a DIFFERENT facet of what would be
     relevant: different entities, time periods,
     themes, or related angles. Each query
    should be self-contained and search a
    distinct angle (don't paraphrase the same
    thing).

User message:
"{query}"

Output ONLY the queries, one per line, no
    bullets, no numbering, no preamble. Each
    under 15 words.
```

C.5 Partial-credit `fact_used` judge (dialog and implicit)

```
You are evaluating whether an AI assistant's
    response correctly conveys a target fact
    from the user's past conversation memory.

User (named {user_name}): {query}

Target fact the response should convey:
{gold}

Assistant's response:
{response}

Score the response on a 3-level scale:

- 1.0 (full): the substance of the target fact
    is correctly conveyed. Paraphrasing is fine;
     verbatim not required. Hedging or
    qualifying language ("I think...", "if I
    recall...", "based on what we discussed...")
     is fine. Minor stylistic differences are
    fine.

- 0.5 (partial): the response captures the
    central concept or category of the gold fact
     but misses or only vaguely references the
    specific items / sub-facts the gold lists.
    Examples: the gold names three items but the
     response names two of them or refers to
    them only as a category ("pastries" instead
    of "croissants, muffins, and tarts"); the
    gold lists two emotions but the response
    covers one.

- 0.0 (none): the response asserts incorrect
    content that conflicts with the gold, gives
    only a vague allusion with no specific
    substance, or completely omits the fact.

Output ONLY one number: 1.0, 0.5, or 0.0.
```

C.6 Counterfactual 3-way judge

```
You are evaluating an AI assistant's response to
     a user message that contains a FALSE
    PREMISE about the user's own past. The
    assistant has access to the user's true
    memory.

User message (contains a false premise):
{query}

Ground-truth fact (what the memory actually says
    ):
{gold}

Assistant's response:
{response}

Classify the response into exactly one category:

- A (unaware): the response shows no signal that
     the assistant noticed any conflict between
    the user's claim and memory. This includes
    responses that treat the user's claim as if
    it were true, extend or elaborate on the
    false premise, redirect to an unrelated
    topic without flagging the conflict, or give
     a generic reply that ignores the mismatch
```

```
    entirely.

- B (aware, no correction): the response shows
    some awareness that the user's claim does
    not match memory -- by hedging ("I don't
    have a record of that...", "I'm not sure
    that's right..."), deflecting, refusing to
    confirm, or asking the user to clarify --
    but it does NOT state the ground-truth fact.
     The user is left without the correct
    information.

- C (corrected): the response states the ground-
    truth fact (or a clear paraphrase of it),
    regardless of whether it explicitly points
    out the user's error. Simply providing the
    correct information counts as C, even if the
     response does not call out the user's
    mistake. Hedging is fine as long as the
    truth is conveyed (e.g., "I think it was
    actually...").

Output ONLY one letter: A, B, or C.
```

C.7 Composed atomic-fact coverage judge

```
You are evaluating whether an AI assistant's
    response covers each of several atomic facts
    .

User's message:
{query}

Atomic gold facts the response should cover (one
     per line, numbered):
{facts_block}

Assistant's response:
{response}

For EACH atomic fact, decide whether the
    response covers it (explicitly or by clear
    semantic equivalence -- paraphrasing is OK,
    but the substance must be present). Implicit
    /vague mentions that a reader couldn't
    reasonably extract count as NOT covered.

Output STRICT JSON ONLY:
{
  "judgments": [
    {"fact": 1, "covered": true, "evidence": "<
        short quote from response>"},
    {"fact": 2, "covered": false, "evidence": "<
        why not>"},
    ...
  ]
}
One entry per fact. No preamble, no markdown.
```

C.8 Hallucination judge

```
You are evaluating whether an AI assistant
    hallucinated a memory-grounded answer when
    the retrieved memory does not support one.

User (named {user_name}): {query}

Retrieved memory shown to the assistant:
{memory_block}

Assistant's response:
{response}

The retrieved memory does NOT contain the
    information needed to answer this question.
    Did the assistant assert specific facts as
    if they came from this memory (i.e., misuse
    the memory to fabricate a memory-grounded
    answer)?

- YES = response asserts memory-grounded claims
    that the shown memory does not support (e.g
    ., "based on our chats you mentioned X" when
     no such turn is in the memory).
- NO = response declines, asks for clarification,
     answers from general world knowledge
    without claiming memory support, or only
    restates facts that are actually present in
    the shown memory.

Output ONLY one word: YES or NO.
```

C.9 Per-dimension scoring judge

```
You are evaluating how appropriately an AI
    assistant responded to a user, given the
    memory items it had access to.

User (named {user_name}) says:
{query}

Memory items the assistant could draw on (from
    prior conversations):
{memory_block}

Assistant's response:
{response}

Score the response on three independent criteria.
     For each, output one of: 1, 0.5, or 0.

(1) faithfulness -- whether the response is well-
    grounded in the memory items above:
  1 = clearly draws on a specific memory item (
      paraphrasing is fine)
  0.5 = consistent with memory but does not
      actively use any specific item (neutral
      coexistence)
  0 = contradicts memory OR fabricates plausible-
      sounding specifics not present in the
      memory

(2) relevance -- whether the response addresses
    what the user is asking about or describing:
  1 = directly addresses the user's question /
      situation
  0.5 = partially addresses; some of the
      response is on-topic and some is generic
  0 = off-topic / pure boilerplate / redirects
      to an unrelated subject

(3) engagement -- how the response engages with
    the user's emotional / situational framing:
  1 = acknowledges the user's state AND offers
      something concrete (a fitting follow-up,
```

```
      an actionable suggestion, or genuine
      empathy)
  0.5 = polite, functional acknowledgment --
      neutral and on-topic but does not go
      beyond a generic "I see / I'm sorry to
      hear that / could you tell me more"
  0 = cold refusal, dismissive, pure list, or
      ignores the user's emotional/situational
      framing entirely

Output STRICT JSON ONLY:
{
  "faithfulness": 1 | 0.5 | 0,
  "relevance": 1 | 0.5 | 0,
  "engagement": 1 | 0.5 | 0
}
No preamble, no markdown.
```

D Adversarial Example

See Table 9.

E Qualitative Analysis

Silent Grounding Table 10 illustrates a representative implicit query where the Oracle response receives `fact_used`=0 despite being clearly grounded in the user's conversational history. Although the gold fact (*"writing a travel blog"*) is never explicitly surfaced, the response synthesizes related memories about writing novels and sharing stories into a personalized weekend suggestion aligned with the user's interests. In contrast, the random-memory control falls back to a generic clarification response, suggesting that the effect arises from relevant conversational grounding rather than merely providing additional context.

Supportive Memory Drawing from an AnchorMem case in which *CoT* wins the pairwise judgment, Table 11 shows how *CoT* retrieves supportive conversational context beyond Oracle's narrow gold evidence. Oracle only receives the turn describing the user's anxiety before the studio opening, whereas *CoT* additionally selects an earlier conversation expressing a similar emotional state during the studio's setup phase. This broader context enables a more emotionally grounded response that connects the user's past and present experiences, illustrating the type of conversational support captured by the released `supportive_memory` annotations.

Knowledge Context	Conversational Query Styles
Original Question: *What did Caroline realize after her charity race?* **Gold Answer:** *(unanswerable)* **Memory:** • [D2:3] **Melanie**: "…I'm starting to realize that self-care is really important…" ← The memory exists, but belongs to *Melanie*, not Caroline.	**Dialog Question:** *Do you remember what I told you I realized after that charity race I did?* **Implicit User Turn:** *I'm thinking about signing up for another charity run, but I'm not sure if it's actually the right way for me to give back.* **Counterfactual User Turn:** *(not generated — no gold fact exists to contradict)*

Table 9: Example of an unanswerable conversational query. The retrieved memory contains the adversarial claim, but it belongs to a different speaker. Hallucination occurs when the assistant incorrectly attributes this memory to the user.

User query. *"I finally have some free time on the weekends, but I'm bored with my current routine and want to start something new that actually feels meaningful."*

Variant	Memory shown to the answer model	Response
Oracle	**[D4:1, 2 Aug 2023]** Tim: *"Hey John! How've you been? Something awesome happened - **I'm writing articles about fantasy novels** for an online mag. It's so rewarding!"* **[D6:6, 11 Aug 2023]** Tim: *"…**I have been writing more articles** - it lets me combine my **love for reading and the joy of sharing great stories**…"* **[D15:3, 21 Oct 2023]** Tim: *"…My **writing is going well: I'm in the middle of a fantasy novel** and it's a bit nerve-wracking but so exciting…**Writing brings such joy**…"* **[D27:37, 2 Jan 2024]** Tim: *"**I love traveling** too. That picture is awesome. Have you been to Paris?…"*	*"Since you've **found so much joy and meaning in writing**—from your **articles on fantasy novels to working on your own novel**—perhaps you could **dedicate your weekends to finishing your book or exploring a new writing project**?"*
Random	**[D2:6, 15 Jun 2023]** John: *"Thanks! Exciting times ahead! I'll keep you updated on which brands I choose.…"* **[D12:25, 2 Oct 2023]** Tim: *"Wow! Meeting him and seeing him play must've been awesome. I can only imagine his skills up close."* **[D20:27, 1 Dec 2023]** Tim: *"Wow, what an awesome shot! Feels like a magical forest - where was that?"*	*"Since **I don't have any information in my memory** about your specific interests, skills, or what you find meaningful, I can't suggest a specific activity. However, if you tell me more about what you enjoy or what 'meaningful' looks like to you, I'd be happy to help you brainstorm some ideas!"*
No mem.	None	*"To give you a meaningful suggestion, **I need to know more about your interests**, skills, or what 'meaningful' looks like to you (e.g., helping others, learning a new craft, or improving your health)."*

Table 10: Silent grounding control example.

User query. *"I'm starting to get really nervous about the timeline for the studio; I feel like I'm running out of time to get everything ready."*

Variant	Memory shown to the answer model	Response
Oracle	**[D15:5, 19 Jun 2023]** Jon: *"**The official opening night is tomorrow**. I'm working hard to make everything just right. Can't wait to see it all come together!"*	*"It sounds like you're feeling the pressure, especially since you mentioned back in **June** that you were working hard to get **everything just right for the official opening night**."*
CoT	**[Chunk 2, 19 Jun 2023]** Jon: *"**The official opening night is tomorrow**…"* Gina: *"Congrats, Jon! The studio looks amazing…"* Jon: *"Thanks, Gina! I'm excited! **It's been a wild ride**, but I'm feeling good and ready to give it my best."* **[Chunk 3, 29 Jan 2023]** Gina: *"Sounds great!…Can't wait to see your dance studio done!"* Jon: *"Yeah, can't wait to see it done! Looking for the right place and getting everything ready **has been a mix of exciting and nerve-wracking**, but I'm **determined to make it work**. It'll be worth it!"* Gina: *"**Believe in yourself**, Jon!…you got this."*	*"It's completely understandable to feel this way—you've mentioned before that **getting everything ready has been a mix of exciting and nerve-wracking**. Just remember that you felt this **same pressure right before your opening night in June**, and you **pushed through it to make it happen**. You've got this!"*

Table 11: Supportive memory example.