CONTRACT MEMORY COMPILER: RESOLVE, THEN TRAVERSE

Anonymous authors
Paper under double-blind review

ABSTRACT

External memory lets language-model agents answer questions about histories too long for the answer model's context window. Updates create a harder problem than retrieving a recent fact: changing one relation can redirect a multi-hop question to records about an entity absent from the question. We study this *update-dependent evidence selection* problem and introduce the Contract Memory Compiler (CMC). Before seeing a question, CMC uses a language model to identify relations in the history and record where each one was stated. It applies later updates to determine the current relations, follows them from entities named in the question, and passes the corresponding original records to the answer model in one call. Thus the current state determines which evidence is read, rather than merely refreshing values in a previously selected context. To the best of our knowledge, CMC achieves state-of-the-art multi-hop accuracy on FactConsolidation, reaching 78.25% overall and 61.0% at 262K. With the extracted relations and answer model held fixed, selecting evidence before resolving updates reduces multi-hop accuracy to 21.50%. We also introduce MQuAKE-MemStream, a derived dataset of ordered memory streams built from MQuAKE-Remastered counterfactual cases.

1 INTRODUCTION

A language model agent may interact with the same user for months. It learns a preference in one session and a decision in another. The user may later revise that decision. The model cannot keep the entire history in its active context, and long inputs can still hide evidence in the middle (Liu et al., 2024). External memory stores information across sessions. It supplies a small set of records for each response. The agent must therefore record useful information, track changes, and find the right records later. Benchmarks for persistent memory test reasoning across sessions, knowledge updates, selective forgetting, and retrieval (Wu et al., 2025; Hu et al., 2026b).

Memory systems address different parts of this process. Compression and synthesis turn interactions into compact, searchable records (Liu et al., 2026). Linked notes and graph memories preserve connections among events and entities (Wu et al., 2026b; Ong et al., 2025). Temporal memory systems retain past versions of facts and identify the current version (Banerjee et al., 2026; Cui et al., 2026). Some agents choose storage, update, and retrieval operations as part of their workflow (Yu et al., 2026b). These designs determine what persists and how it can be accessed. At answer time, the model sees only the records placed in its current input. We study how to choose those records after the stored facts change. An update can change which older record is needed, even when the question stays the same.

Consider a long interaction history. An early record places a weekly meeting in Room A. Separate records place Room A on floor 2 and Room B on floor 5. Much later, an update moves the meeting to Room B. For the question "On which floor is the weekly meeting?", the relevant evidence is now the move and the earlier record about Room B. Before the update, the same question required the record about Room A. The question and both floor records are unchanged. Only the current meeting room has changed, yet that change determines which earlier record the assistant must read.

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.

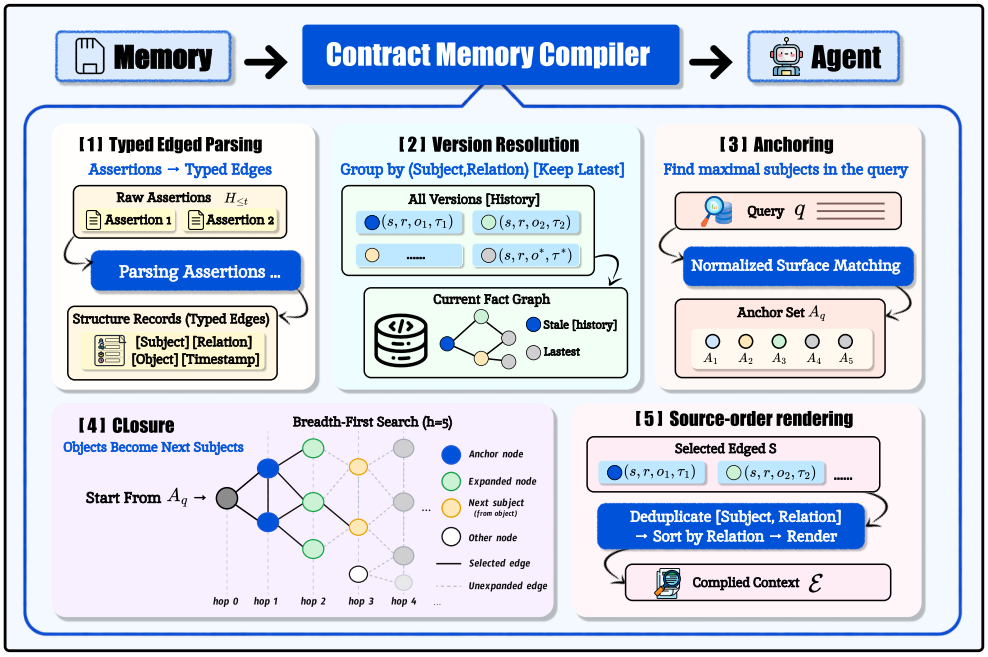

Figure 1: The Contract Memory Compiler between the assertion memory and the reader. Typed-edge parsing and version resolution build one current object per subject–relation key. Anchoring, closure, and source-order rendering run once per query: explicit subjects start a five-hop traversal in which each object becomes the next subject, and the selected source records are rendered for one reader call.

The memory must therefore resolve the current room and use it to select the next record. If selection precedes the update, the context can contain Room A's floor while omitting Room B's. Refreshing the meeting-room fact inside that context cannot recover the omitted floor record. Similarity search may find the move but miss the floor record because Room B is absent from the question. Iterative retrieval can issue another query after identifying Room B (Trivedi et al., 2023). Our setting instead selects evidence for one answer call. We call this task *update-dependent evidence selection*. It requires the current relation to guide record selection before the reader receives its context.

The Contract Memory Compiler (CMC) implements this order. It prepares the memory before questions arrive (Figure 1). It uses a language model to extract simple relationships from the records. It stores each relationship with the position of its source record. It then processes the relationships in source order. For each subject and relation, it keeps the latest value. In the example, CMC identifies Room B as the current meeting room. The older note about Room B's floor remains available. When the question arrives, CMC starts from the meeting named in the question. It follows the current location to Room B. It then reads Room B's floor. The answer model receives the original move and floor notes. It answers once.

We evaluate this read order on FactConsolidation, a MemoryAgentBench dataset for selective forgetting. It has long, ordered histories with contradictory updates. Its questions ask about the final memory state (Hu et al., 2026b). On questions that require several hops, CMC achieves 78.25% substring exact match overall and 61.0% on 262K token histories. To the best of our knowledge, these results establish the state of the art for FactConsolidation multi-hop questions.

To test update-dependent selection on controlled multi-hop chains, we construct MQuAKE-MemStream from the counterfactual cases of MQuAKE-Remastered (Zhong et al., 2025). This derived dataset turns released facts before and after an edit into ordered memory streams. It provides a second controlled test of evidence selection after updates.

Our contributions are:

- We define a bounded read contract. It specifies how updates affect current values and evidence coverage.
- We implement the contract in CMC. It extracts relations before questions arrive, resolves updates in order, and selects original records for one answer call.
- We construct MQuAKE-MemStream, a derived dataset for multi-hop memory reading based on MQuAKE-Remastered counterfactual cases. It provides ordered histories and controls for version resolution and evidence selection.

2 READING A SERIALLY OVERWRITTEN MEMORY

Setting. Let $H_{\leq t}$ be an ordered assertion stream. A parsed assertion is an edge $e = (s, r, o, \tau)$, where s is its subject, r its relation, o its object, and τ its source position. Relations follow last-write-wins semantics: a later assertion overwrites an earlier one with the same key (s, r). The current state contains one edge per key,

$$G_t = \left\{ \underset{e \in \mathcal{E}_t:\, e.\text{key}=k}{\arg\max}\ e.\tau : k \in \text{keys}(\mathcal{E}_t) \right\},$$

where $\mathcal{E}_t$ is the set of parsed edges. Recency is defined by source order in this setting.

The read contract. For a query q, let A_q be its explicitly named graph subjects and let $d_{G_t}(A_q, s)$ be the shortest directed distance from any anchor to s. For hop limit h, the selected evidence is

$$C_h(q, G_t) = \{(s, r, o, \tau) \in G_t : d_{G_t}(A_q, s) < h\}.$$

This gives two checkable properties of the selected interpreted edges. *Currency*: every edge in C_h is the last-written parsed edge for its key. *Bounded graph coverage*: every current outgoing edge in G_t from a subject reachable at depth less than h is included. We call this set a bounded closure. It follows graph reachability and may include branches beyond the relation requested by the query. Its usefulness depends on the supporting facts being parsed and reachable within h hops of an explicit anchor. These graph properties do not assign versions to unresolved source text retained alongside the selected edges.

Updates can change which keys a read needs. In the running example, the meeting moves from Room A to Room B while both rooms keep their floors. Before the update, a read contains meeting $\rightarrow$ Room A and Room A $\rightarrow$ floor 2. Replacing only the values of these previously selected keys yields meeting $\rightarrow$ Room B and Room A $\rightarrow$ floor 2: both edges are current, but Room B's floor is absent. Recomputing reachability in G_t selects meeting $\rightarrow$ Room B and Room B $\rightarrow$ floor 5. The second lookup must change from Room A's floor to Room B's floor. This motivates resolution before traversal. Our experiments separately test value choice within fixed keys and key selection before value refresh.

Reader interface. The compiler renders source records for C_h together with unresolved records as evidence E for a language-model reader $R(q, E)$ under a fixed input window. All CMC contexts in the shared direct-extraction comparison fit its window. Selecting among competing closures when the window binds remains a separate extension of this read interface.

3 CONTRACT MEMORY COMPILER

CMC takes an ordered assertion history and prepares evidence for its reader (Figure 1). Its preparation stage extracts relations, their source positions, and unresolved original records, then materializes a current graph shared across queries about that history. For each question, it finds explicit subject anchors, traverses current outgoing edges up to five hops, and renders the supporting source records. Version resolution, anchoring, traversal, and rendering are deterministic.

Integration boundary and lifecycle. This preparation runs before query-time reading. The selected facts are returned as original text, together with unresolved records, for one frozen-reader generation. An empty graph selection uses the original history under the same evidence budget. The internal loop expands a graph frontier: each selected object identifies a subject to inspect next. Algorithm 1 gives the shared-LLM procedure evaluated below.

Stage 1: parse and retain source provenance. The frontend receives source records and a catalogue of relation names from previous extraction batches. It assigns each record one status: `facts`, containing directed subject–relation–object triples; `unresolved`, retaining the record for direct reading; or `no_fact`, contributing neither an edge nor a residual. Structural validation checks record identities, the JSON schema, and literal subject and object spans. Source-only LLM alignment proposes a common name for equivalent relation slots. Fixed case, punctuation, and possessive normalization produces entity keys. Accepted edges retain the source position and original text; unresolved records form the residual set U. Offline extraction, alignment, and repair use source text alone (Appendix D.1). The grammar adapter uses manually specified sentence patterns for this stage.

Stage 2: materialize the current graph. Accepted assertions are replayed in source order into a dictionary keyed by normalized subject and aligned relation. Each occurrence replaces that key's stored edge, leaving other keys unchanged. The resulting G_t is shared across queries about the history; original assertions remain in the source. For example, "[4] The weekly meeting is in Room B" produces $(\texttt{weekly_meeting}, \texttt{held_in}, \texttt{room_b}, 4)$, storing $(\texttt{room_b}, 4)$ at key $(\texttt{weekly_meeting}, \texttt{held_in})$. Traversal follows this value to Room B, and rendering returns record 4 and the original record stating Room B's floor.

Stage 3: anchor explicit query entities. The compiler normalizes the question and finds graph subjects that occur as whole normalized text spans. If one matched subject is contained in a longer matched subject, it keeps the longer match. All remaining matches form A_q. This is a deterministic surface-form rule; entity disambiguation and alias inference are outside this implementation.

Stage 4: traverse the current graph. The question determines the initial subject anchors. From these anchors, breadth-first traversal includes every current outgoing edge at subject depths zero through four. Each selected object supplies a subject for the next expansion, so a revised value redirects subsequent evidence selection. The result is a bounded, potentially branching subgraph. Shortest-depth bookkeeping handles cycles, and edges are deduplicated by (s, r). Expansion stops when the frontier is empty or the five-hop cap is reached. Relation choice is left to the reader, which receives the selected original assertions.

Stage 5: render source records. For a nonempty selected subgraph, CMC retrieves the original numbered records supporting its edges and adds all unresolved source records. An empty selection uses the original history. The renderer deduplicates records by source position, restores source order, and applies a 60,000-character budget including the evidence prefix. On overflow, it retains the newest contiguous suffix of whole records. The reader receives the evidence and question with the instruction that larger serial numbers are newer. All selected evidence fits in the primary shared-extraction comparison. Currency and bounded coverage apply to the selected parsed edges; unresolved text retains its original wording.

Additional memory interfaces. The broader MemoryAgentBench evaluation (Hu et al., 2026b) includes tasks adapted from LongMemEval (Wu et al., 2025), ReDial (Li et al., 2018), ∞Bench (Zhang et al., 2024), and DetectiveQA (Xu et al., 2024), alongside document QA and classification tasks. We use original-window retrieval for accurate retrieval and recommendation, card/raw lookup for classification and DetectiveQA, and coverage notes with original-text fallbacks for summarization. The window interface uses 2,048-character windows with 256-character overlap, BM25 selection, and source-order rendering. Dataset origins and interface details are in Appendix I.

Implementation cost. Latest-state construction is one ordered pass over the frozen extraction. A graph read scans keys for anchors, traverses the bounded subgraph, and sorts selected records by

source position. At 262K, the measured CPU read takes 13.8 ms median (144.3 ms p95); Appendix J reports the timing protocol. Reader tokens include the chat template; extraction cost is separate.

4 EXPERIMENTS

We first test whether resolving updates before selecting evidence improves answers on FactConsolidation. Its 800 questions comprise equal single- and multi-hop sets across 6K, 32K, 64K, and 262K histories; we use the released substring exact-match score. The primary comparison fixes the extracted facts, answer model, and prompt, changing only the records placed in the answer context. We then vary the reader and extraction method, test MQuAKE-MemStream, and report broader MemoryAgentBench results. Implementation and evaluation protocols are in Appendices C, D, F, and I.

4.1 FACTCONSOLIDATION: SELECTING EVIDENCE AFTER UPDATES

CMC's preparation stage processes all 18,336 unique source statements without seeing questions or answer labels. We freeze the extracted relations and unresolved records for all parsed views in Table 1. Every view uses the same source histories, HY4-preview reader, prompt, and 60,000-character evidence cap; only the evidence presented to the reader changes.

Table 1: FactConsolidation with shared extraction and one HY4-preview reader. SH and MH are accuracy (%); Tokens is mean reader-input length; Trunc. counts clipped contexts.

View	SH	MH	Tokens	Trunc.
	Untruncated views			
CMC	97.25	**78.25**	185	0
Prior read + refresh	97.25	21.50	195	0
Stale closure	32.00	11.75	182	0
Fact BM25	**97.50**	20.75	1,528	0
	Budgeted (600/800 truncated)			
Latest state	53.00	17.00	12,351	600
Raw history	27.50	5.00	12,870	600

Prior read + refresh chooses records using previous relation values and then replaces the selected values with current ones. Stale closure keeps CMC's selected relations but uses previous values. Fact BM25 ranks current facts by question overlap. CMC and these three controls fit the input limit. Latest state and raw history omit records on 600 questions; their scores include truncation. Figure 2 separates history lengths. Full protocols and extraction costs are in Appendix D.

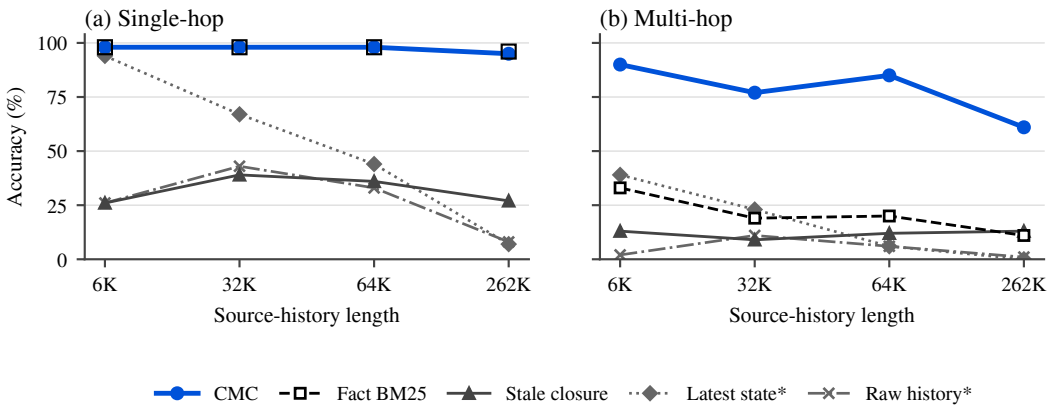

Figure 2: FactConsolidation accuracy by history length. Each point covers 100 questions with the same HY4-preview reader. Latest-state and raw-history scores include truncation on 600 of 800 questions.

Selecting after updates. Prior read + refresh chooses records from previous relation values and then refreshes those records. All presented parsed facts are current, but newly needed records can be absent. CMC ties this control at 97.25% single-hop and scores 78.25% versus 21.50% multi-hop. With extraction and reader fixed, the 56.75-point gap isolates record selection (Appendix G).

Retrieving current facts. CMC and fact BM25 are nearly tied on single-hop questions (97.25% and 97.50%): the BM25 pool contains the scored answer string on all 389 single-hop questions CMC answers correctly. The gap appears on multi-hop questions (78.25% versus 20.75%), where an intermediate entity can be absent from the question. In one 262K counterfactual history, the question asks for the origin of the sport played by Christian Abbiati. CMC follows Abbiati → cornerback → field hockey → Philippines; BM25 retrieves the Abbiati fact but misses the record giving field hockey's origin. Across the 400 multi-hop questions, CMC is correct on 233 items where BM25 is wrong, versus three in the other direction. On 218 of those 233 items, BM25 lacks at least one CMC-selected fact; on 109, it lacks the scored answer string. Neither read is truncated, and CMC uses fewer mean reader-input tokens (185 versus 1,528). These coverage diagnostics show where question-based ranking often loses the newly relevant next hop.

Evidence selection at matched length. We also give BM25 the same per-question evidence budget as CMC, packing whole current facts from a fixed top-100 ranking. Mean reader inputs are then 182 versus 185 tokens. Single-hop scores remain close (97.00% versus 97.25%), but BM25 reaches only 7.25% multi-hop versus CMC's 78.25% (Figure 3). On the 286 multi-hop questions answered only by CMC, the matched BM25 pool never contains every CMC-selected fact and contains the scored answer string on just 19. The difference persists when the reader receives nearly equal amounts of text. Appendix E gives the packing rule and results by history length.

Appendix Table 9 compares graph selection with lexical and dense ranking over the same extracted current facts. BGE-M3 dense retrieval reaches 33.25% multi-hop accuracy with 1,535 mean reader-input tokens, above fact BM25's 20.75% but below CMC's 78.25% with 185 tokens. Changing the similarity function improves retrieval, but still leaves many next-hop records unnamed by the question.

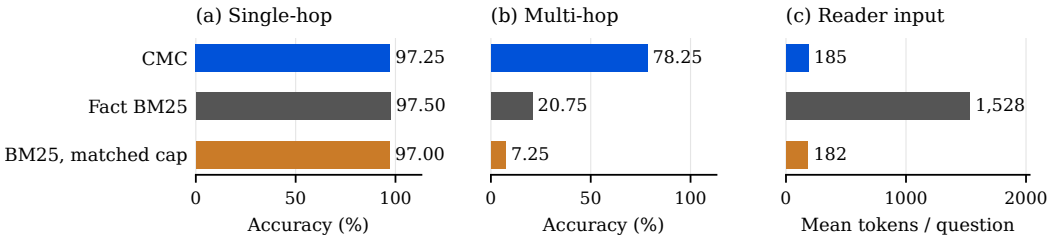

Figure 3: FactConsolidation at matched evidence length. BM25 packs whole current facts within each question's CMC budget. Panel (c) shows mean HY4-preview reader-input tokens over all 800 questions.

Version choice also matters. Stale closure keeps CMC's selected relations but substitutes previous values where available; it reaches 11.75% multi-hop with 182 mean input tokens. This isolates the cost of following old values from the earlier control's cost of choosing records before resolution. Raw history and the latest-state view reach 5.00% and 17.00% under the shared cap, but both omit records on 600 questions. Their scores therefore reflect truncation as well as evidence selection.

Traversal depth also affects coverage: multi-hop accuracy is 5.50, 54.00, 71.75, and 78.25% at caps of one, two, three, and five hops, respectively (Appendix H).

Query-time conflict resolution. We adapt a released post-retrieval conflict-resolution pipeline (Reddy and Reddy Challaram, 2026) to the same 800 histories and questions. With HY4-preview, it reaches 98.00% single-hop and 60.75% multi-hop, versus CMC's 97.25% and 78.25%. CMC gains 79 multi-hop answers and loses nine. Each system retains its own extraction and call sequence,

so this is an end-to-end comparison rather than a fixed-extraction control. Appendix D.4 gives the adaptation and call costs.

4.2 Reader and extraction changes

We first replace the HY4-preview answer model while keeping the LLM extraction and evidence packets fixed. With Qwen3.5-4B, CMC reaches 76.50% multi-hop versus 20.50% for fact BM25; with Llama-3.1-8B, it reaches 55.00% versus 11.50%. Single-hop scores remain close within each reader (Table 2). Thus the multi-hop evidence advantage persists across these readers, although their absolute scores differ.

We separately replace the LLM extraction with a grammar adapter. Under Qwen3.5-4B and Qwen3.5-9B, CMC reaches 79.00% and 80.25% multi-hop, versus 22.50% for each reader given the untruncated current state. With Qwen3.5-4B, a size-matched random negative control that excludes CMC's selected keys scores 1.25% multi-hop. These rows test a different extraction and control from the reader-swap rows; Appendix B gives the full grammar grid. With the same HY4-preview reader and CMC read, direct LLM extraction reaches 78.25% multi-hop accuracy, close to the grammar adapter's 79.75%, while deriving relation names from source records rather than manually specified patterns (Appendix D.1).

Table 2: FactConsolidation reader/frontend settings, 800 questions each. The grammar rows share one extraction and use untruncated current state as control. The LLM rows share the extracted facts and original reader messages; the control is fact BM25. Each reader uses its own chat interface.

Reader	Frontend	CMC SH	CMC MH	Ctrl. SH	Ctrl. MH
	Grammar adapter; control = full current state				
Qwen3.5-4B	Grammar	97.25	**79.00**	95.25	22.50
Qwen3.5-9B	Grammar	**98.00**	**80.25**	94.25	22.50
	Shared LLM extraction; control = fact BM25				
Qwen3.5-4B	Shared LLM	96.50	**76.50**	96.50	20.50
Llama-3.1-8B	Shared LLM	90.75	**55.00**	92.00	11.50
HY4-preview	Shared LLM	97.25	**78.25**	**97.50**	20.75

4.3 MQuAKE-MemStream: a constructed memory dataset

We construct MQuAKE-MemStream from the two-hop counterfactual cases of MQuAKE-Remastered (Zhong et al., 2025; 2023). It converts released pre- and post-edit facts into ordered memory streams while retaining the source questions and answer labels. Every method receives the answer-bearing support facts, so this is a controlled test of version resolution and evidence selection rather than the original model-editing task. The transformation, pooling, and consistency audit are in Appendix F.

Table 3: MQuAKE-MemStream, constructed from the two-hop MQuAKE-Remastered cases. Frozen Qwen3.5-4B reader; 800 single-hop and 798 multi-hop questions. Raw history uses its newest suffix; the last two rows are percentage-point differences from CMC.

Context given to the reader	Single hop	Multi hop
CMC	**99.63**	**96.62**
Latest state	98.75	53.38
Stale closure	30.88	13.41
Raw history (windowed)	89.63	60.53
CMC − latest state	+0.88	+43.23
CMC − stale closure	+68.75	+83.21

With the pinned Qwen3.5-4B reader (Table 3), CMC reaches 96.62% multi-hop accuracy, versus 53.38% when the reader sees all current facts and 13.41% when the selected relations use earlier values. Selecting a connected read from the current state gains 43.23 points over presenting that state in full. Resolving values within the selected relations gains 83.21 points over stale values.

Table 4: MemoryAgentBench. Upper blocks copy published Table 3 numbers as reference; their readers differ and they are not bolded against this work. The last block holds one HY4-preview reader. AR and Rec are original-window retrieval. SF is current-graph traversal on overwritten assertions; the two cells are the official 262K split (200 questions). MCC, Summary, and DetectiveQA are the original same-reader campaign, not Algorithm 1: card/raw lookup on MCC and DetectiveQA, coverage notes on Summary, and the registered JSON-field DetQA decoder. Bold is the highest value *inside the last block* of each column.

	AR: original windows				*MCC*	**Rec: windows**	*LRU*		**SF: current graph**	
System	SH-QA	MH-QA	LME(S*)	EventQA	MCC	Rec.	Sum	DetQA	FC-SH	FC-MH
Published long-context (reference)										
GPT-5-mini (400K)	85.0	71.0	63.3	78.2	84.0	13.2	56.3	76.1	78.0	28.0
Published retrieval (reference)										
BM25	66.0	56.0	45.3	74.6	75.4	13.6	19.0	52.1	48.0	3.0
Text-Embed-3-Small	60.0	44.0	48.3	63.0	70.0	15.3	17.7	54.9	28.0	3.0
Text-Embed-3-Large	54.0	44.0	50.3	70.0	72.4	16.2	18.2	56.3	28.0	4.0
Qwen3-Embedding-4B	57.0	47.0	43.3	71.4	78.0	12.2	14.8	59.2	29.0	3.0
RAPTOR	29.0	38.0	34.3	45.8	59.4	12.3	13.4	42.3	14.0	1.0
GraphRAG	47.0	47.0	35.0	34.4	39.8	9.8	0.4	39.4	14.0	2.0
MemoRAG	29.0	33.0	20.0	56.0	77.0	13.1	9.2	50.7	21.0	7.0
HippoRAG-v2	76.0	66.0	50.7	67.6	61.4	10.2	14.6	57.7	54.0	5.0
Mem0	25.0	32.0	36.0	37.5	32.4	10.0	4.8	36.6	18.0	2.0
Cognee	31.0	26.0	29.3	26.8	35.4	10.1	2.3	29.6	28.0	3.0
Zep	44.0	25.0	38.3	42.5	62.8	12.1	4.2	28.2	7.0	3.0
Self-RAG	35.0	42.0	25.7	31.8	11.6	12.8	0.9	35.2	19.0	3.0
MemGPT	41.0	38.0	32.0	26.2	67.6	14.0	2.5	42.3	28.0	3.0
MIRIX	62.0	61.0	37.3	29.8	38.4	9.8	9.9	40.8	14.0	2.0
MIRIX (4.1-mini)	73.0	75.0	51.0	53.0	61.0	10.3	18.9	62.0	20.0	3.0
Same HY4-preview reader										
HY4-preview (suffix)	18.0	19.0	6.0	75.4	**86.4**	13.1	14.3	76.1	8.0	1.0
HY4 BM25	78.0	60.0	39.3	85.6	85.8	14.1	14.8	77.5	30.0	2.0
HY4 dense+BM25	69.0	44.0	37.3	87.2	86.2	11.9	**15.2**	**78.9**	28.0	1.0
CMC + task interfaces	**85.0**	**77.0**	**44.3**	**90.8**	86.2	**17.8**	10.2	67.6	**95.0**	**61.0**

4.4 Same-reader MemoryAgentBench

Table 4 reports the full MemoryAgentBench grid (Hu et al., 2026b). Only SF uses the current-graph read; direct comparisons use the four local HY4-preview rows.

At the 262K FactConsolidation split, CMC reaches 95.0% single-hop and 61.0% multi-hop, versus 8.0% and 1.0% for the suffix and 30.0% and 2.0% for same-reader BM25. These cells are a subset of the 800-question comparison above.

CMC leads the same-reader rows on five retrieval and recommendation columns and is similar on classification (85.8–86.4), but trails dense+BM25 on summarization (10.2 versus 15.2) and DetectiveQA (67.6 versus 78.9). Task protocols and diagnostics are in Appendix I.

5 Related Work

Writing and maintaining agent memory. Memory Networks and RAG distinguish stored material from what is read for a task (Weston et al., 2015; Miller et al., 2016; Lewis et al., 2020). Long-term dialogue adds multi-session recall and invalidation (Xu et al., 2022; Bae et al., 2022). Generative Agents retrieve experiences, and MemGPT moves context across memory tiers (Park et al., 2023; Packer et al., 2023). SimpleMem and Mem0 compress dialogue (Liu et al., 2026; Chhikara et al., 2025); A-Mem revises linked notes (Xu et al., 2025). RMM builds summaries at multiple scales and refines retrieval using cited evidence (Tan et al., 2025). AgeMem learns storage and update actions (Yu et al., 2026b); THEANINE retains older events in timelines (Ong et al., 2025). Our question is which original records enter one answer context after a current relation changes.

Current state and graph retrieval. APEX-MEM organizes temporally grounded events in a property graph and uses an agent to resolve changing information at query time (Banerjee et al.,

2026). MemTxn checks whether writes have source support and selects visible versions of conflicting facts (Cui et al., 2026). Zep and SodaMem represent changing relationships in temporal graphs (Rasmussen et al., 2025; Wan et al., 2026). GAM separates event progression from longer-term topic consolidation (Wu et al., 2026b). MAGMA traverses semantic, temporal, causal, and entity graphs (Jiang et al., 2026a). HippoRAG uses a knowledge graph and personalized PageRank for multi-hop retrieval (Gutiérrez et al., 2024); Mnemis combines similarity search with hierarchical graph selection (Tang et al., 2026). GraphRAG summarizes graph communities for corpus-level questions; RAPTOR retrieves through a summary tree (Edge et al., 2024; Sarthi et al., 2024). ReFind searches raw dialogue turns (Li et al., 2026). A stage-wise comparison of graph and non-graph memories likewise finds that implementation choices affect measured performance (Hu et al., 2026a). We isolate a narrower property: under single-valued, ordered assertions, the resolved object determines which source record a bounded traversal selects next.

Multi-hop evidence and answer assembly. IRCoT interleaves reasoning steps with new retrieval requests when later evidence depends on earlier findings (Trivedi et al., 2023). MeLLo handles multi-hop questions after knowledge edits, while MQuAKE-Remastered studies how edits affect downstream answers (Zhong et al., 2023; 2025). SCM routes queries among lexical, graph, and iterative memory reads (Tobkin and Yang, 2026). A post-retrieval assembly method separates evidence extraction from the final answer policy (Reddy and Reddy Challaram, 2026). CMC studies a different boundary: it resolves updates and selects source records before a single answer call. Our fixed-extraction controls compare this order of operations; they do not rank complete adaptive pipelines against one another. The G-Walk and MeLLo ports in Appendix B.2 receive a resolved fact bank, so that comparison holds version choice fixed. MQuAKE-MemStream reads supplied support facts rather than measuring parametric editing.

6 Discussion

The multi-hop gain comes from selecting connected current facts: refreshing previously selected values loses 56.75 points (Section 4.1). The LLM frontend avoids hand-written patterns, but exact anchoring, missing links, and the hop limit restrict coverage. FactConsolidation came from a broader development grid, so independent confirmation remains needed. An automated screen flags 33 of 800 reference answers as possibly superseded; all remain in the official scores. The multi-hop gap persists under strict normalized exact match and is at least 50.50 points under adversarial rescoring of the 26 flagged multi-hop items (Appendix B.3).

7 Conclusion

CMC resolves current relations before it selects evidence. On ordered, single-valued assertions, that order lets an overwrite redirect the next hop: refreshing the previously selected graph reaches 21.50% multi-hop accuracy, and traversing the current graph reaches 78.25%. The same gap appears against retrieval over the current facts and under two additional readers. The demonstrated gain is for ordered, single-valued assertion streams.

Reproducibility Statement

We record inputs, model identifiers, prompts, settings, outputs, and native scores, and verify reused responses before joint scoring. Appendix D documents the evaluation protocol. The supplementary package includes the compiler, evaluation scripts, MQuAKE-MemStream data, and per-example predictions.

AI Use Statement

AI coding assistants assisted implementation, experiment execution, artifact checks, and manuscript editing. The authors retain responsibility for the methods, results, citations, and final manuscript.

REFERENCES

Sanghwan Bae, Donghyun Kwak, Soyoung Kang, Min Young Lee, Sungdong Kim, Yuin Jeong, Hyeri Kim, Sang-Woo Lee, Woomyoung Park, and Nako Sung. Keep me updated! memory management in long-term conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3769–3787, 2022. doi: 10.18653/v1/2022.findings-emnlp.276. URL https://aclanthology.org/2022.findings-emnlp.276/.

Pratyay Banerjee, Masud Moshtaghi, Shivashankar Subramanian, Amita Misra, and Ankit Chadha. APEX-MEM: Agentic semi-structured memory with temporal reasoning for long-term conversational AI. In *Proceedings of ACL*, pages 16470–16489, 2026. URL https://aclanthology.org/2026.acl-long.749/.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, 2020. doi: 10.18653/v1/2020.nlp4convai-1.5. URL https://aclanthology.org/2020.nlp4convai-1.5/.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.137. URL https://aclanthology.org/2024.findings-acl.137/.

Jiayi Chen, Yingcong Li, and Guiling Wang. MemFlow: Intent-driven memory orchestration for small language model agents. *arXiv preprint arXiv:2605.03312*, 2026. URL https://arxiv.org/abs/2605.03312.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. In *ECAI 2025*, 2025. doi: 10.3233/FAIA251160. URL https://doi.org/10.3233/FAIA251160.

Hanshuai Cui, Zhiqing Tang, Zhi Yao, Fanshuai Meng, Qianli Ma, and Weijia Jia. MemTxn: A transaction boundary for source-supported updates and complete-state recovery in agent memory. *arXiv preprint arXiv:2607.27834*, 2026. URL https://arxiv.org/abs/2607.27834.

Xin Ding, Xinrui Wang, Yifan Yang, Hao Wu, Shiqi Jiang, Qianxi Zhang, Liang Mi, Hanxin Zhu, Kun Li, Yunxin Liu, Zhibo Chen, and Ting Cao. MemCompiler: Compile, don't inject—state-conditioned memory for embodied agents. *arXiv preprint arXiv:2605.07594*, 2026. URL https://arxiv.org/abs/2605.07594.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph RAG approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024. URL https://arxiv.org/abs/2404.16130.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6ddc001d07ca4f319af96a3024f6dbd1-Abstract-Conference.html.

Zexue He, Yu Wang, Churan Zhi, Yuanzhe Hu, Tzu-Ping Chen, Lang Yin, Ze Chen, Tong Arthur Wu, Siru Ouyang, Zihan Wang, Jiaxin Pei, Julian McAuley, Yejin Choi, and Alex Pentland. MemoryArena: Benchmarking agent memory in interdependent multi-session agentic tasks. *arXiv preprint arXiv:2602.16313*, 2026. URL https://arxiv.org/abs/2602.16313.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *Proceedings of the First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.06654.

Sen Hu, Yuxiang Wei, Jiaxin Ran, Xueran Han, Zhiyuan Yao, Huacan Wang, Ronghao Chen, and Lei Zou. Does memory need graphs? a unified framework and empirical analysis for long-term dialog memory. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26758–26782, 2026a. doi: 10.18653/v1/2026.acl-long.1232. URL https://aclanthology.org/2026.acl-long.1232/.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. In *International Conference on Learning Representations*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/fd1eff9dd295df50a41f2521942fa31d-Abstract-Conference.html.

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

Dongming Jiang, Yi Li, Guanpeng Li, and Bingzhe Li. MAGMA: A multi-graph based agentic memory architecture for AI agents. In *Proceedings of ACL*, pages 36848–36865, 2026a. URL https://aclanthology.org/2026.acl-long.1709/.

Eric Hanchen Jiang, Zhi Zhang, Yuchen Wu, Levina Li, Dong Liu, Xiao Liang, Rui Sun, Yubei Li, Edward Sun, Haozheng Luo, Zhaolu Kang, Aylin Caliskan, Kai-Wei Chang, and Ying Nian Wu. Memory as a controlled process: Learned adaptive memory management for LLM agents. *arXiv preprint arXiv:2607.13591*, 2026b. URL https://arxiv.org/abs/2607.13591.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 1311–1316, 2019. doi: 10.18653/v1/D19-1131. URL https://aclanthology.org/D19-1131/.

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, pages 9459–9474, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230-Abstract.html.

Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. Towards deep conversational recommendations. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/800de15c79c8d840f4e78d3af937d4d4-Abstract.html.

Ruizhe Li, Licheng Zhang, Benfeng Xu, Mingxuan Du, Zheren Fu, and Weidong Chen. When your agent opens the chat app: Agent-controlled search over raw chat logs rivals structured memory. *arXiv preprint arXiv:2608.12888*, 2026. URL https://arxiv.org/abs/2608.12888.

Xin Li and Dan Roth. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002. URL https://aclanthology.org/C02-1150/.

Yidan Lin, Kaixiang Wang, Jiong Lou, and Jie Li. Stop when memory suffices: Evidence-conditioned progressive execution for LLM agents. *arXiv preprint arXiv:2608.01285*, 2026. URL https://arxiv.org/abs/2608.01285.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. SimpleMem: Efficient lifelong memory for LLM agents. *arXiv preprint arXiv:2601.02553*, 2026. URL https://arxiv.org/abs/2601.02553.

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

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. Benchmarking natural language understanding services for building conversational agents. *arXiv preprint arXiv:1903.05566*, 2019. URL https://arxiv.org/abs/1903.05566.

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

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1409, 2016. doi: 10.18653/v1/D16-1147. URL https://aclanthology.org/D16-1147/.

Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seung-won Hwang, Dongha Lee, and Jinyoung Yeo. Towards lifelong dialogue agents via timeline-based memory management. In *Proceedings of NAACL*, pages 8631–8661, 2025. URL https://aclanthology.org/2025.naacl-long.435/.

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

Egor Pakhomov, Erik Nijkamp, and Caiming Xiong. ConvoMem benchmark: Why your first 150 conversations don't need RAG. *arXiv preprint arXiv:2511.10523*, 2025. URL https://arxiv.org/abs/2511.10523.

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*, 2023. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, Avishek Anand, and Adam Jatowt. It's high time: A survey of temporal question answering. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28852–28881, 2026. doi: 10.18653/v1/2026.acl-long.1332. URL https://aclanthology.org/2026.acl-long.1332/.

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

Vikas Reddy and Sumanth Reddy Challaram. Reliable post-retrieval assembly for agent memory: Separating evidence extraction from policy execution. *arXiv preprint arXiv:2606.01435*, 2026. URL https://arxiv.org/abs/2606.01435.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8a2acd174940dbca361a6398a4f9df91-Abstract-Conference.html.

Yiting Shen, Kun Li, Wei Zhou, and Songlin Hu. Mem2ActBench: A benchmark for evaluating long-term memory utilization in task-oriented autonomous agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8173–8190. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.370. URL https://aclanthology.org/2026.acl-long.370/.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, Anand Rajan Iyer, Tianlong Chen, Huan Liu, Chen-Yu Lee, and

Tomas Pfister. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8416–8439, 2025. doi: 10.18653/v1/2025.acl-long.413. URL https://aclanthology.org/2025.acl-long.413/.

Zihao Tang, Xin Yu, Ziyu Xiao, Zengxuan Wen, Zelin Li, Jiaxi Zhou, Hualei Wang, Haohua Wang, Haizhen Huang, Weiwei Deng, Feng Sun, and Qi Zhang. Mnemis: Dual-route retrieval on hierarchical graphs for long-term LLM memory. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23914–23928, 2026. doi: 10.18653/v1/2026.acl-long.1096. URL https://aclanthology.org/2026.acl-long.1096/.

Joshua Tobkin and David Yang. Supra Cognitive Modes: A routed architecture for agent memory. *arXiv preprint arXiv:2607.19096*, 2026. URL https://arxiv.org/abs/2607.19096.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, 2023. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Sergey Volkov, Yang Li, and Ye Luo. StateFuse: Deterministic conflict-preserving memory for multi-agent systems. *arXiv preprint arXiv:2607.05844*, 2026. URL https://arxiv.org/abs/2607.05844.

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

Xinyu Wang, Mingze Li, Peng Lu, Xiao-Wen Chang, Lifeng Shang, Jinping Li, Fei Mi, Prasanna Parthasarathi, and Yufei Cui. InfMem: Learning system-2 memory control for long-context agent. *arXiv preprint arXiv:2602.02704*, 2026. URL https://arxiv.org/abs/2602.02704.

Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1410.3916.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/d813d324dbf0598bbdc9c8e79740ed01-Abstract-Conference.html.

Di Wu, Zixiang Ji, Asmi Kawatkar, Bryan Kwan, Jia-Chen Gu, Nanyun Peng, and Kai-Wei Chang. LongMemEval-V2: Evaluating long-term agent memory toward experienced colleagues. *arXiv preprint arXiv:2605.12493*, 2026a. URL https://arxiv.org/abs/2605.12493.

Zhaofen Wu, Hanrong Zhang, Fulin Lin, Wujiang Xu, Xinran Xu, Yankai Chen, Henry Peng Zou, Shaowen Chen, Weizhi Zhang, Xue Liu, Philip S. Yu, and Hongwei Wang. GAM: Hierarchical graph-based agentic memory for LLM agents. In *Proceedings of ACL*, pages 34647–34664, 2026b. URL https://aclanthology.org/2026.acl-long.1600/.

Jing Xu, Arthur Szlam, and Jason Weston. Beyond goldfish memory: Long-term open-domain conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022. URL https://aclanthology.org/2022.acl-long.356/.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-Mem: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/19909c36f51abc4856b4560aff3d36d6-Abstract-Conference.html.

Zhe Xu, Jiasheng Ye, Xiaoran Liu, Xiangyang Liu, Tianxiang Sun, Zhigeng Liu, Qipeng Guo, Linlin Li, Qun Liu, Xuanjing Huang, and Xipeng Qiu. DetectiveQA: Evaluating long-context reasoning on detective novels. *arXiv preprint arXiv:2409.02465*, 2024. URL https://arxiv.org/abs/2409.02465.

Tianyu Yang, Sudipta Paul, Vijay Srinivasan, Vivek Kulkarni, and Srinivas Chappidi. TRUSTMEM: Learning trustworthy memory consolidation for LLM agents with long-term memory. *arXiv preprint arXiv:2606.25161*, 2026. URL https://arxiv.org/abs/2606.25161.

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. MemAgent: Reshaping long-context LLM with multi-conv RL-based memory agent. In *International Conference on Learning Representations*, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/4264ee4376776907c0b87ed70b959585-Abstract-Conference.html.

Yi Yu, Liuyi Yao, Yuexiang Xie, Qingquan Tan, Jiaqi Feng, Yaliang Li, and Libing Wu. Agentic memory: Learning unified long-term and short-term memory management for large language model agents. In *Proceedings of ACL*, pages 21457–21483, 2026b. URL https://aclanthology.org/2026.acl-long.981/.

Haozhen Zhang, Haodong Yue, Tao Feng, Quanyu Long, Jianzhu Bao, Bowen Jin, Weizhi Zhang, Xiao Li, Jiaxuan You, Chengwei Qin, and Wenya Wang. Learning query-aware budget-tier routing for runtime agent memory. *arXiv preprint arXiv:2602.06025*, 2026. URL https://arxiv.org/abs/2602.06025.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞Bench: Extending long context evaluation beyond 100k tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15262–15277, 2024. doi: 10.18653/v1/2024.acl-long.814. URL https://aclanthology.org/2024.acl-long.814/.

Shaochen Zhong, Yifan (Louie) Lu, Lize Shao, Bhargav Bhushanam, Xiaocong Du, Yixin Wan, Yucheng Shi, Daochen Zha, Yiwei Wang, Ninghao Liu, Kaixiong Zhou, Shuai Xu, Kai-Wei Chang, Louis Feng, Vipin Chaudhary, and Xia Ben Hu. MQuAKE-Remastered: Multi-hop knowledge editing can only be advanced with reliable evaluations. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f782860c2a5d8f675b0066522b8c2cf2-Abstract-Conference.html.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15686–15702, 2023. doi: 10.18653/v1/2023.emnlp-main.971. URL https://aclanthology.org/2023.emnlp-main.971/.