CoRelMem: Coherent Relation Induction for Long-Term Agent Memory

Anonymous authors
Paper under double-blind review

Abstract

Long-term memory is essential for LLM-based agents to accumulate and reuse information across extended interactions. Existing memory systems commonly retrieve historical information based on lexical or semantic similarity, implicitly assuming that memories more similar to the query are also more useful for answering it. However, semantic similarity does not necessarily reflect reasoning relevance. Relevant memories with weak query similarity may be overlooked, although their relevance may be revealed through associations with other retrieved memories. Meanwhile, highly similar but low-value memories may receive excessive priority and occupy the limited retrieval budget. To address this issue, we propose CoRelMem, a structured coherent relation induction framework for long-term agent memory. CoRelMem organizes long-term memories through logical association and temporal evolution, enabling retrieval to follow cross-memory dependencies and recover relevant memories that cannot be identified through direct semantic matching alone. Based on these structures, a knowledge-guided dual-path retrieval strategy expands candidate memories from complementary relational and temporal perspectives, improving the coverage of useful information. CoRelMem further introduces an information-aware reranking mechanism that considers both retrieval relevance and conditional information contribution, reducing the priority of superficially similar but less informative memories. Experiments on LoCoMo and LongMemEval demonstrate consistent improvements in both memory retrieval and downstream question answering. The code is available at https://anonymous.4open.science/r/CoRelMem-DC8B.

1 Introduction

As LLM-based agents increasingly engage in long-term interactions, effective memory has become essential for retaining and reusing information from historical conversations (Hu et al., 2026; Huang et al., 2026; Chhikara et al., 2025). However, continuously growing interaction histories cannot be directly included in the context due to both computational cost and irrelevant information (Packer et al., 2023; Fang et al., 2026; Xu et al., 2026b). Existing long-term memory methods therefore typically extract important information into memory units and retrieve a small subset for each new query. Most retrieval methods rely on lexical or semantic similarity, implicitly assuming that memories more similar to the query are also more useful for answering it (Zhong et al., 2024; Chhikara et al., 2025; Xu et al., 2026c).

However, semantic similarity does not necessarily reflect reasoning relevance (Zhang et al., 2025b; 2026e). As illustrated in Fig. 1, a memory that shares many keywords or semantic concepts with the query may still provide little useful evidence, while a less similar memory may contain the information required for answering the question. More importantly, such weakly matched evidence may be connected to already retrieved memories through shared entities or relational links, making it accessible through structured memory organization even when its direct query similarity is low (Anokhin et al., 2025; Rasmussen et al., 2025).

This phenomenon gives rise to two closely related challenges in long-term memory retrieval. On the one hand, **relevant memories with low query similarity can be overlooked**. Similarity-based retrieval tends to miss memories that are only weakly matched to the query but can be discovered through their relations with other relevant memories. This issue is particularly prominent in multi

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.

hop reasoning, where a highly similar memory often provides only a partial clue and the complete evidence must be further recovered through cross memory associations or event evolution (Anokhin et al., 2025; Su et al., 2026). On the other hand, **highly similar but low-value memories may receive excessive priority**. Even when relevant information has entered the candidate set, similarity alone may fail to distinguish truly informative memories from superficially similar distractors. Such high-similarity distractors can occupy the limited retrieval budget and interfere with downstream answer generation (Zhang et al., 2025b; Fang et al., 2026; Xu et al., 2026b).

Our empirical analysis on LoCoMo further supports this observation. As shown in Fig. 1(a), memories that contain the information required to answer the query do not always receive higher similarity scores than irrelevant but semantically similar memories, and the two distributions show substantial overlap. Fig. 1(b) further shows that adding such highly similar irrelevant memories causes a larger drop in question answering performance than adding randomly selected irrelevant memories. These results indicate that semantic similarity alone is insufficient to reliably identify useful memories, motivating the need to both recover weakly matched relevant memories and reduce the influence of highly similar but low-value ones.

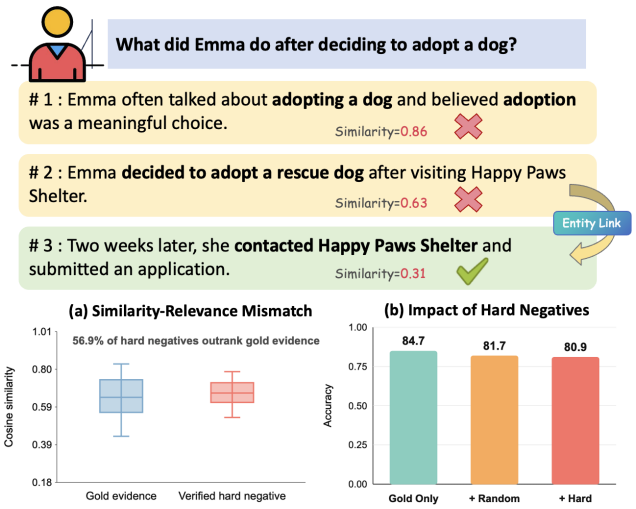


Figure 1: Illustration and empirical analysis of the limitations of similarity-based retrieval in long-term memory.

To address these challenges, we propose CoRelMem, a structured long-term memory framework designed to improve both the discovery and prioritization of relevant memories. To recover evidence that may be weakly matched to the query, CoRelMem introduces a coherent relation induction strategy that organizes cross-memory logical associations and temporal evolution through a memory association graph and temporal trajectories. During retrieval, seed memories are expanded through a knowledge-guided dual-path retrieval strategy over the memory association graph and temporal trajectories, enabling additional relevant evidence to be discovered beyond direct semantic matching. To further distinguish truly informative memories from weakly useful candidates introduced during structural expansion, we introduce an information-aware reranking mechanism that jointly considers retrieval relevance and conditional information contribution.

Together, these components form a unified framework for discovering broader evidence and selecting more informative memories. Our main contributions are summarized as follows. ❶ We propose CoRelMem, a structured long-term memory framework that employs coherent relation induction to organize cross-memory logical associations and temporal evolution, facilitating the discovery of relevant memories. ❷ We develop a knowledge-guided dual-path retrieval strategy and an information-aware reranking mechanism to improve evidence coverage and prioritize truly informative memories. ❸ Extensive experiments on LoCoMo and LongMemEval demonstrate consistent improvements in both memory retrieval and downstream question answering.

2 Related Work

Existing agent memory methods mainly differ in how they organize and retrieve historical information. In management, flat-memory methods such as Mem0 store relatively independent facts or summaries (Chhikara et al., 2025), while hierarchical approaches such as LightMem organize memories across multiple stages or granularities to improve storage and access efficiency (Fang et al., 2026). Structured methods such as A-MEM further make cross-memory relations explicit through evolving associative networks (Xu et al., 2026c), with other approaches exploring relational, temporal, causal, or event-centric structures (Anokhin et al., 2025; Rasmussen et al., 2025; Zhang et al.,

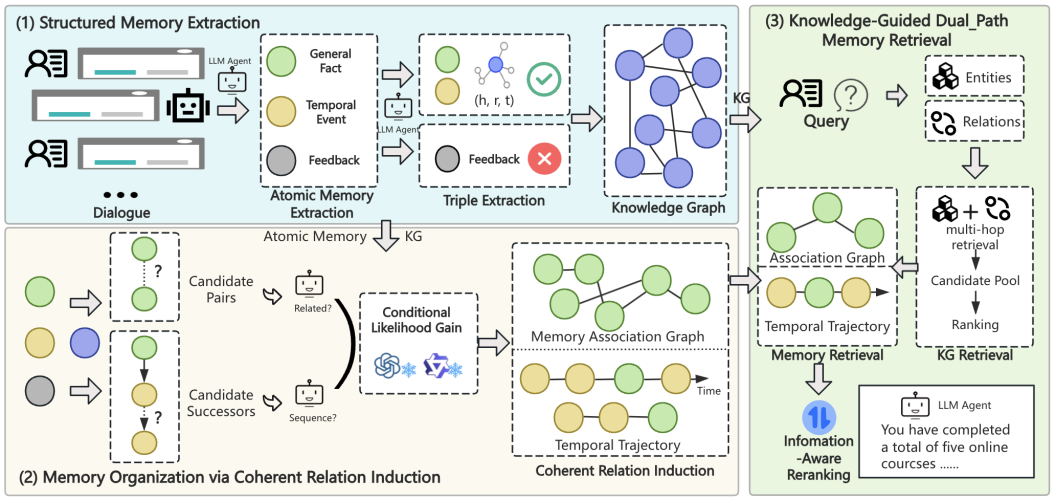


Figure 2: Overall framework of CoRelMem. The framework consists of three stages: Structured Memory Extraction, Coherent Relation Induction, and Knowledge-Guided Dual-Path Memory Retrieval.

2025b; Gao et al., 2025; Wu et al., 2026b; Zhang et al., 2026e). These studies demonstrate that explicit memory structures can connect dispersed information and provide useful support for retrieving related evidence. In retrieval, semantic similarity remains a common mechanism for candidate selection, while recent methods increasingly incorporate adaptive and structure-aware strategies. MemGAS, for example, organizes memories at multiple granularities and uses an entropy-based router to dynamically select the retrieval granularity before applying LLM-based filtering (Xu et al., 2026b). Other approaches further explore retrieval-budget control, graph expansion, and temporal constraints (Zhang et al., 2026a; 2025c; Wang et al., 2026; Anokhin et al., 2025; Zhang et al., 2025b; Rasmussen et al., 2025; Yang et al., 2026; Su et al., 2026). Overall, memory retrieval is gradually moving beyond isolated similarity matching toward better utilization of memory organization to uncover more relevant evidence. Detailed related work is provided in Appendix A.1.

3 Methodology

The overall framework of CoRelMem is illustrated in Fig. 2. CoRelMem first transforms raw interaction histories into structured atomic memories through Structured Memory Extraction, where memories are categorized into *General Fact*, *Temporal Event*, and *Feedback*. Informative memories are further converted into triples to establish a structured knowledge representation. Building on these memory units, Coherent Relation Induction organizes cross-memory dependencies from two complementary perspectives, constructing a memory association graph to capture logical relations and temporal trajectories to model temporal evolution. During retrieval, query-relevant seed memories are identified and expanded through the learned memory structures, after which Information-Aware Reranking selects the most informative evidence for final answer generation.

3.1 Structured Memory Extraction

Raw dialogues generated during long-term interactions often contain redundant and low-value information. Directly converting all such content into structured memory is neither necessary nor efficient, since transient or weakly informative memories may yield unreliable triples and introduce substantial computational overhead for subsequent relation modeling. We therefore first decompose the dialogue history into atomic memories and filter out low-value content before constructing the base knowledge representation.

Given a historical dialogue sequence $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$, an LLM extracts semantically complete and relatively independent atomic memories, yielding $\mathcal{M} = \{m_1, m_2, \ldots, m_N\}$. These memories are further classified into three categories according to their roles in long-term mem-

ory: *General Fact*, *Temporal Event*, and *Feedback*, denoted by $\mathcal{M}_G$, $\mathcal{M}_T$, and $\mathcal{M}_F$, respectively, where $\mathcal{M} = \mathcal{M}_G \cup \mathcal{M}_T \cup \mathcal{M}_F$. Detailed definitions for these three memory types are provided in Appendix A.3.1. Feedback memories are excluded from subsequent memory organization.

Entities and relational triples are then extracted from the retained memories. For each memory m_i, its entity set $\mathcal{E}_i$ is first identified, and relations are extracted only between entity pairs grounded in the same memory, producing triples of the form (h, r, t), where h, r, and t denote the head entity, relation, and tail entity, respectively. The extracted triples collectively form a base knowledge graph $\mathcal{G}_K = (\mathcal{E}, \mathcal{R})$. Each triple retains the index of its source memory, establishing an explicit mapping between entity-level knowledge and atomic memories.

3.2 Memory Organization via Coherent Relation Induction

While the base knowledge graph captures explicit relations at the entity level, long-term reasoning often relies on dependencies between memories themselves. Such dependencies may arise from complementary information about the same experience or from continuous state transitions over time. To capture these memory-level dependencies, Coherent Relation Induction (CRI) organizes memories through logical associations and temporal evolution. These two structures are represented by a memory association graph and temporal trajectories, respectively. A theoretical interpretation of CRI and its two complementary memory structures is provided in Appendix A.2.

To provide a unified measure of information dependency between memories, Conditional Likelihood Gain (CLG) is defined as

$$\Delta(y; x \mid C) = \ell(y \mid C, x) - \ell(y \mid C), \tag{1}$$

$$\ell(y \mid x) = \frac{1}{|y|} \log P_\theta(y \mid x), \tag{2}$$

where $\ell(\cdot)$ denotes the length-normalized conditional log-likelihood computed by a frozen language model. Intuitively, $\Delta(y; x \mid C)$ measures how much the predictability of target y improves after introducing information x under context C.

3.2.1 Memory Association Graph Construction

The memory association graph captures direct dependencies beyond semantic similarity. Candidate pairs are first identified through shared entities and locally reachable relations in the knowledge graph, with semantic neighbors added to complement potentially missing associations. An LLM then determines whether each candidate pair (m_i, m_j) exhibits a meaningful direct relation. To suppress spurious connections, CLG further measures the information dependency between candidate memories. For the direction $m_i \to m_j$, the conditional information gain is defined as

$$g_{i \to j} = \Delta(m_j; m_i \mid C_{ij}), \tag{3}$$

where C_{ij} denotes the local context of the candidate pair. The bidirectional gains are aggregated into the association strength

$$S_{ij}^A = f\left(g_{i \to j}, g_{j \to i}\right), \tag{4}$$

where $f(\cdot)$ is a symmetric aggregation function. An association edge (m_i, m_j) is retained only when the LLM judgment is positive and $S_{ij}^A > \tau_A$. The resulting graph is denoted as $\mathcal{G}^A = (\mathcal{M}, \mathcal{E}^A)$, where edges represent coherent dependencies between memories.

3.2.2 Temporal Trajectory Construction

Temporal memories may exhibit explicit state evolution that cannot be captured by non-directional associations alone. Simply ordering events by timestamp is insufficient, since temporally adjacent memories may belong to different evolving processes. We therefore organize related temporal memories into temporal trajectories.

Let $\mathcal{M}^T$ denote the temporal memory set. For each memory m_i^T, candidate successors are selected from later events satisfying $t_j > t_i$, with the base knowledge graph used to remove unrelated candidates. Each candidate pair (m_i^T, m_j^T) is then examined by two successive LLM judgments. The first determines whether the two memories belong to the same evolving event thread, while the second

verifies whether m_j^T forms a direct state transition from m_i^T. For the remaining candidates, CLG further measures their transition dependency:

$$g_{i\to j}^T = \Delta(m_j^T; m_i^T). \tag{5}$$

Only transitions with positive information gain are retained. When multiple valid successors exist, their gains are normalized to select the most likely successor, resulting in time-ordered trajectories $m_1^T \to m_2^T \to \cdots \to m_L^T$. The resulting temporal trajectory set is denoted by $\mathcal{T} = \{T_1, T_2, \ldots, T_P\}$, where each T_p represents a coherent sequence of temporal memories. Detailed implementation procedures for both memory association graph construction and temporal trajectory construction are provided in Appendix A.3.2.

3.3 Knowledge-Guided Dual-Path Memory Retrieval

Structured memory organization is further exploited during retrieval to uncover relevant evidence beyond semantic similarity. Given a query q, semantic relevance $S_{\text{sem}}(q, m)$ and structural relevance $S_{\text{stru}}(q, m)$ are jointly used to rank memories, where the latter is obtained through multi-hop matching over the base knowledge graph $\mathcal{G}_K$. The ranking score is defined as

$$S_{\text{seed}}(m \mid q) = (1-\lambda) S_{\text{sem}}(q, m) + \lambda S_{\text{stru}}(q, m), \tag{6}$$

where λ balances semantic and structural relevance. The top-ranked memories form the direct candidate set $\mathcal{C}_D$, while the highest-ranked subset is selected as the seed set $\mathcal{S}_q \subset \mathcal{C}_D$ for subsequent structural expansion.

3.3.1 Dual-Path Retrieval

For the memory association graph $\mathcal{G}^A$, Personalized PageRank is used to propagate query relevance from seed memories to their associated neighbors. Let $\mathbf{v}$ denote the personalized restart distribution derived from the query relevance of the seed memories. The propagation is performed by

$$\mathbf{p}^{(t+1)} = \alpha \mathbf{v} + (1-\alpha)\mathbf{P}^\top \mathbf{p}^{(t)}, \tag{7}$$

where $\mathbf{P}$ is the normalized transition matrix and α is the restart probability. Propagation is restricted to an H_A-hop local subgraph around the seeds, producing the association candidate set $\mathcal{C}_A$.

For temporal trajectories $\mathcal{T}$, query-relevant contiguous segments are searched within trajectories containing seed memories. Given $\mathcal{T} = (e_1, \ldots, e_L)$ and event relevance $r_i = S_{\text{sem}}(q, e_i)$, the relevance of a segment $P_{a:b}$ is defined as

$$R_T(P_{a:b} \mid q) = \eta \max_{a \le i \le b} r_i + (1-\eta)\frac{1}{b-a+1}\sum_{i=a}^{b} r_i. \tag{8}$$

The segment length is limited by L_T, and the selected segments form the temporal candidate set $\mathcal{C}_T$. Finally, the direct candidates and structurally expanded candidates are merged as $\mathcal{C}_q = \mathcal{C}_D \cup \mathcal{C}_A \cup \mathcal{C}_T$, providing complementary evidence from direct matching, cross-memory associations, and temporal evolution. Implementation details are provided in Appendix A.3.3.

3.3.2 Information-aware Reranking

Structural expansion may introduce weakly relevant candidates, so Information-Aware Reranking (IAR) is applied to prioritize evidence that is both relevant and informative for the current query. For each candidate U, its information contribution is measured by

$$S_{\text{info}}(q, U) = \Delta(q; U). \tag{9}$$

After separately normalizing semantic relevance and information gain, the final reranking score is computed as

$$S_{\text{IAR}}(q, U) = (1-\beta) S_{\text{sem}}(q, U) + \beta S_{\text{info}}(q, U), \tag{10}$$

where β balances relevance and information contribution. Candidates are reranked according to S_{IAR}, and the Top-K memories are retained as the final context for answer generation. Implementation details are provided in Appendix A.3.4.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Datasets CoRelMem is evaluated on two long-term memory benchmarks, LoCoMo (Maharana et al., 2024) and LongMemEval-S (Wu et al., 2025), to assess its effectiveness under long interaction histories and complex memory reasoning scenarios. LoCoMo contains long-range multi-session conversations with corresponding question answering tasks, covering multi-hop reasoning, temporal reasoning, open-domain questions, and single-hop reasoning. LongMemEval-S consists of long interaction histories constructed for individual test questions and covers single-session information, preference, multi-session reasoning, knowledge update, and temporal reasoning. Detailed dataset statistics and question-type distributions are provided in Appendix A.4.1.

Baselines and Metrics Four representative long-term agent memory methods are selected as baselines, including Mem0, A-Mem, MemGAS, and LightMem. These methods cover representative paradigms for memory storage, organization, and retrieval, enabling comparison across different long-term memory management strategies. For question answering evaluation, Accuracy, F1, BLEU, and ROUGE are adopted as automatic metrics. F1 measures content overlap between predicted and reference answers, while BLEU and ROUGE evaluate lexical and phrase-level correspondence. In addition, Accuracy is used to evaluate the consistency between model outputs and reference answers, complementing traditional text-matching metrics for open-ended responses. Detailed descriptions of the baseline methods and evaluation metrics are provided in Appendix A.4.3 and Appendix A.4.2, respectively.

Implementation Experiments are conducted using Qwen3-4B-Instruct-2507 (Yang et al., 2025a) and Llama-3.2-3B-Instruct (Grattafiori et al., 2024) as the LLM backbones. Our implementation is mainly based on the open-source LightMem (Fang et al., 2026) codebase, and all methods are compared under the same backbone and data split. Models are deployed with vLLM on a single NVIDIA A800 80 GB GPU. Following the LightMem setting, LLMLingua-2 (Pan et al., 2024) is used for memory compression with a compression rate of 0.6, and all-MiniLM-L6-v2 is used to generate 384-dimensional memory embeddings. By default, we set $\lambda = 0.1$, $\tau_A = 0.6$, $\alpha = 0.2$, $H_A = 2$, and $L_T = 3$. We use Top-10 seed memories, a candidate pool of 60, and a final Top-10 retrieval budget. LightMem, A-Mem, Mem0, and CoRelMem use the same embedding model, while MemGAS retains its original Contriever-based retrieval setting. Detailed hyperparameter ranges are provided in the appendix A.4.4.

4.2 OVERALL RESULTS

Table 1 compares CoRelMem with representative long-term memory methods on LoCoMo. CoRelMem achieves the best overall performance under both Qwen3-4B-Instruct-2507 and Llama-3.2-3B-Instruct across ACC, F1, BLEU, and ROUGE metrics, demonstrating consistent effectiveness across different backbone models. The improvements are particularly evident on multi-hop and temporal reasoning tasks, where answering requires either connecting distributed memories or tracking information over time. CoRelMem also performs strongly on open-domain and single-hop questions, indicating that structured memory organization preserves the ability to retrieve direct factual information. As shown in Table 2, CoRelMem also achieves the best overall performance on LongMemEval-S under Qwen3-4B, further validating its generalizability across different long-term memory benchmarks.

Table 3 further evaluates retrieval quality at different retrieval sizes. Under Qwen3-4B, CoRelMem consistently achieves the best Recall, MRR, and nDCG across all K, showing that it can retrieve more relevant memories while ranking them more effectively. Under Llama3.2-3B, CoRelMem achieves the best MRR and nDCG across all settings and the best Recall@3, while MemGAS obtains higher Recall at larger K. This suggests that CoRelMem is particularly effective at prioritizing relevant memories near the top of the ranking rather than relying only on a larger retrieval set. Together with the downstream QA improvements in Table 1, these results indicate that relation-aware memory organization and information-aware reranking improve both retrieval quality and the usefulness of the retrieved memories for subsequent reasoning.

Table 1: Overall performance on LoCoMo. The best results are shown in bold, and the second-best results are underlined.

Model	Method	Overall						Multi Hop	Temp.	Open Domain	Single Hop
		ACC	F1	B-1	R-1	R-2	R-L				
Qwen3-4B	Mem0	32.34	13.88	9.18	15.43	5.69	14.78	34.40	33.02	30.21	31.63
	A-Mem	40.45	17.52	10.47	18.45	8.87	17.69	41.49	30.22	31.25	45.07
	MemGAS	52.40	12.38	7.82	13.47	6.05	12.70	49.29	23.68	31.25	66.83
	LightMem	67.53	33.96	20.47	34.15	16.95	32.94	69.86	60.12	52.08	71.34
	Ours	**72.86**	**37.36**	**22.96**	**37.41**	**18.85**	**35.52**	**79.43**	**62.31**	**53.12**	**76.93**
Llama3.2-3B	Mem0	58.90	6.97	3.95	8.42	2.21	7.68	60.64	71.34	67.71	52.56
	A-Mem	53.51	6.07	3.47	6.59	2.14	5.98	52.84	52.96	61.46	53.03
	MemGAS	39.48	9.34	4.98	9.91	4.19	9.20	32.27	13.71	43.75	51.25
	LightMem	67.86	10.12	6.27	10.83	3.56	9.90	68.44	68.85	62.50	67.90
	Ours	**72.34**	**13.28**	**6.41**	**13.74**	**4.84**	**12.67**	**74.11**	**73.52**	**68.75**	**71.70**

Table 3: Retrieval performance on LoCoMo. All metrics are reported as percentages. The best results are shown in bold, and the second-best results are underlined.

Model	Method	$K = 3$			$K = 5$			$K = 10$		
		Recall	MRR	nDCG	Recall	MRR	nDCG	Recall	MRR	nDCG
Qwen3-4B	BM25	66.03	63.34	61.75	72.37	65.02	64.55	82.08	66.57	68.30
	Dense	70.21	68.15	66.04	76.67	69.90	68.99	84.31	70.97	72.24
	MemGAS	49.58	44.03	43.30	60.40	46.42	47.74	76.84	48.47	53.47
	Ours	**77.11**	**77.76**	**75.07**	**81.83**	**78.88**	**77.19**	**87.49**	**79.49**	**79.56**
Llama3.2-3B	BM25	45.37	44.27	42.65	49.66	45.60	44.58	58.76	47.55	48.24
	Dense	48.48	47.37	45.30	53.77	49.26	47.85	59.81	50.50	50.46
	MemGAS	49.58	44.03	43.30	**60.40**	46.42	47.74	**76.84**	48.47	53.47
	Ours	**51.74**	**52.93**	**50.13**	56.79	**54.38**	**52.46**	62.97	**55.53**	**54.99**

4.3 Ablation Study

To examine the contribution of different components, we consider three ablation variants. As shown in Table 4, removing any major component generally degrades performance, indicating that the three components provide complementary benefits. Removing KG weakens entity-level structural cues for candidate discovery and seed localization, with a particularly clear impact on temporal reasoning. Although the knowledge graph is not the final memory representation, it provides useful support for subsequent organization and retrieval. Removing CRI causes a larger drop on cross-memory reasoning tasks, especially multi-hop questions, showing the benefit of explicitly organizing logical associations and temporal evolution across memories. Removing IAR also leads to a noticeable decline, as structural expansion may introduce weakly relevant candidates that occupy the limited context budget. This confirms the role of IAR in prioritizing more informative memories after structural expansion.

Table 2: Performance of Qwen3-4B on LongMemEval-S. Know., Multi., Single., and Temp. denote Knowledge Update, Multi-Session, Single-Session, and Temporal Reasoning, respectively. The best results are shown in bold, and the second-best results are underlined.

Method	Overall	Know.	Multi.	Single.	Temp.
Mem0	44.40	51.28	37.59	59.87	28.57
A-Mem	40.20	55.12	27.81	55.41	25.56
MemGAS	40.40	46.15	24.81	**63.46**	25.56
LightMem	46.20	51.28	46.62	60.90	25.56
Ours	**49.20**	**58.97**	**48.87**	60.26	**30.83**

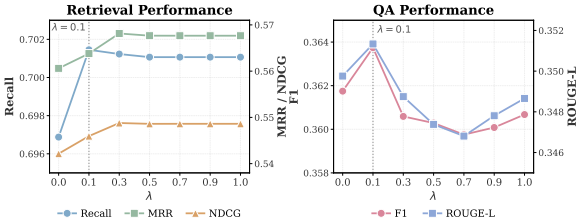
Table 4: Ablation results on LoCoMo.*w/o KG* removes the base knowledge graph and its structural guidance for candidate discovery and retrieval. *w/o CRI* removes Coherent Relation Induction, including both the memory association graph and temporal trajectories. *w/o IAR* removes Information-Aware Reranking, so structurally expanded candidates are no longer prioritized by their information contribution.

Model	Method	Overall						Multi Hop	Temp.	Open Domain	Single Hop
		ACC	F1	B-1	R-1	R-2	R-L				
Qwen3-4B	Ours	**72.86**	**37.36**	**22.96**	**37.41**	18.85	**35.52**	**79.43**	**62.31**	**53.12**	**76.93**
	w/o KG	71.04	37.01	22.82	37.19	**19.01**	35.33	77.30	58.26	50.00	76.22
	w/o CRI	70.65	36.49	22.14	36.54	18.24	34.81	74.82	61.99	43.75	75.62
	w/o IAR	67.59	33.38	19.54	33.39	16.25	32.03	70.56	59.81	50.00	71.58
Llama3.2-3B	Ours	**72.34**	**13.28**	**6.41**	**13.74**	**4.84**	**12.67**	**74.11**	**73.52**	68.75	**71.70**
	w/o KG	70.06	12.40	6.01	13.06	4.71	12.01	73.40	64.80	69.79	70.99
	w/o CRI	70.13	11.93	5.75	12.38	4.31	11.36	71.99	68.85	69.79	70.04
	w/o IAR	69.94	11.33	5.52	11.82	3.92	10.78	73.40	68.22	**79.17**	68.37

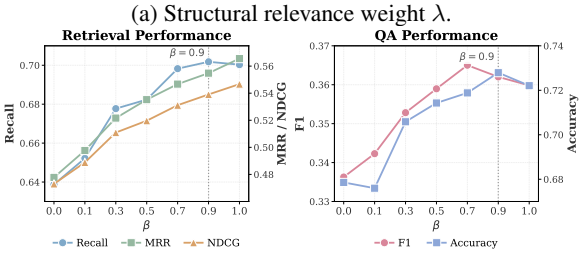
4.4 Hyperparameter Sensitivity

We examine the sensitivity of CoRelMem to three key hyperparameters: the seed-scoring weight λ, the IAR fusion weight β, and the final retrieval size K_R, as shown in Fig. 3. Additional analyses of hyperparameter are provided in Appendix A.5.

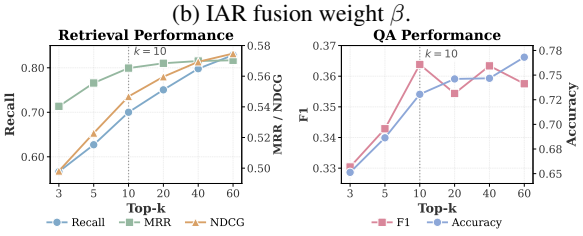
As shown in Fig. 3a, introducing structural relevance improves both retrieval and downstream QA over purely semantic seed selection. Performance is strongest around $\lambda = 0.1$, while larger values provide no consistent gains, indicating that structural signals are most effective as a complement to semantic relevance. Fig. 3b shows a similar trend for IAR, with increasing β generally improving retrieval quality and downstream QA and achieving strong overall performance around $\beta = 0.9$. At $\beta = 1.0$, QA performance slightly declines despite continued gains in some retrieval metrics, suggesting that conditional information contribution remains most effective when combined with the original relevance signal. As shown in Fig. 3c, increasing K_R improves evidence coverage, while the gains in MRR, NDCG, and downstream QA gradually saturate. Larger retrieval sizes recover more relevant memories but may also introduce redundant or weakly relevant evidence.



(a) Structural relevance weight λ.



(b) IAR fusion weight β.



(c) Final retrieval size K_R.

Figure 3: Sensitivity analysis of the key hyperparameters in CoRelMem.

4.5 Case Study

Table 5 highlights the limitation of similarity-based retrieval and the advantage of CoRelMem. For the question *What recipes has Joanna made?*, similarity-based retrieval ranks several recipe-related but weakly informative memories highly, while concrete recipe evidence appears at lower positions.

Table 5: Case study comparing similarity-based retrieval with CoRelMem. Bold text highlights the key information directly relevant to the answer.

Question: *What recipes has Joanna made?*	
Similarity-Based Retrieval	**Ours**
#1 Joanna is revising and perfecting a family recipe that turned out to be tasty.	#3 Joanna is trying dairy-free options . . . ; she has made **dairy-free chocolate coconut cupcakes with raspberry frosting**, . . .
#2 Joanna has been creating desserts that work for various diets, including cookies, pies, and cakes, and plans to share more recipes.	#4 *[Association]* Joanna is revising and perfecting a family recipe, . . . and has created a **cake with strawberries and chocolate** . . .
#8 Joanna tried a new dairy-free recipe with her family . . . and **the dish was presented on a white plate with raspberries.**	#5 *[Trajectory]* Joanna highlights the importance of dairy-free desserts . . . Joanna tried a new dairy-free recipe with her family, . . . **the dish was presented on a white plate with raspberries.**
#9 Joanna expresses enthusiasm to try the dairy-free recipe and emphasizes her love for sweet treats.	#6 Joanna shares a recipe for a **blueberry dessert with a gluten-free crust made using coconut milk**, . . .
#10 Joanna has created a **cake with strawberries and chocolate** as a result of revising old recipes.	#9 Joanna responds to . . . and has revised old recipes to **make a cake with strawberries and chocolate**.

CoRelMem instead promotes more informative memories and organizes them through association paths and temporal trajectories. It recovers specific dishes such as *dairy-free chocolate coconut cupcakes with raspberry frosting* and a *blueberry dessert with a gluten-free crust*, while also linking related recipe and temporal information. This case shows that CoRelMem can improve both the discovery and prioritization of answer-relevant memories.

4.6 Cost Analysis

We further evaluate the performance–cost trade-off on LoCoMo in terms of LLM token consumption and the number of LLM calls. As shown in Fig. 4, CoRelMem achieves the highest overall accuracy while requiring substantially fewer tokens and model calls than Mem0 and A-MEM. This suggests that the structured memory construction and relation induction process does not rely on excessive LLM inference to obtain its performance gains. LightMem remains the most computationally efficient method, but this lower cost comes with reduced accuracy compared with CoRelMem. In contrast, CoRelMem occupies a more balanced region of the performance–cost space, improving answer accuracy while maintaining moderate computational overhead. These results indicate that the additional computation introduced by coherent relation induction is effectively converted into improved memory utility, yielding a favorable trade-off between effectiveness and efficiency.

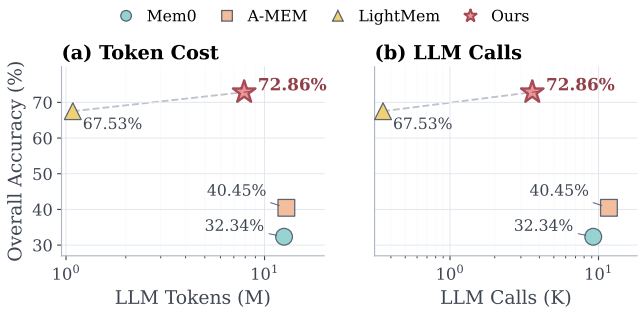


Figure 4: Performance–cost trade-off on LoCoMo in terms of LLM token consumption and LLM calls.

5 Conclusion

This work addresses the limitations of similarity-based retrieval in long-term agent memory by introducing CoRelMem, which organizes memories through logical associations and temporal evolution and exploits these structures for more effective retrieval and reranking. By recovering weakly matched relevant memories and prioritizing candidates with greater information contribution, CoRelMem improves both memory coverage and retrieval quality. Experiments on LoCoMo and LongMemEval demonstrate consistent gains in memory retrieval and downstream question answering, with particularly clear improvements on multi-hop and temporal reasoning tasks. These results highlight the value of explicitly modeling cross-memory relations for complex long-term memory reasoning.

AI USE STATEMENT

Generative AI tools were used to assist with code debugging, literature summarization, language polishing, and manuscript editing. All AI-assisted content was reviewed and verified by the authors. Code-related assistance was validated through actual execution and testing, while literature summaries and revised text were checked against the original sources and the authors' intended meaning. The authors take full responsibility for the final content of this work, including any text, code, or other materials produced with the aid of generative AI.

ETHICS STATEMENT

This study uses publicly available long-term memory benchmark datasets and publicly accessible pretrained language models. It does not involve the recruitment of human participants, the collection of new personally identifiable information, or the creation of new sensitive datasets. All conversations and memory content used in our experiments are drawn from existing public benchmarks and are used solely for research evaluation and analysis. All experiments are conducted entirely in an offline research setting and do not involve real-user data collection, external user profiling, or persistent memory deployment in real-world environments. We do not release any new user data or personally identifiable information as part of this work.

REPRODUCIBILITY STATEMENT

We make several efforts to support the reproducibility of this work. Section 3 describes the proposed CoRelMem framework, while Appendix A.3.1 provides additional methodological details. Section 4.1 specifies the evaluation protocol and major experimental settings, and Appendix A.4 provides further implementation details. Additional theoretical analyses and supplementary experimental results are included in the appendix. The code is available at https://anonymous.4open.science/r/CoRelMem-DC8B.

REFERENCES

Petr Anokhin, Nikita Semenov, Artyom Sorokin, Dmitry Evseev, Andrey Kravchenko, Mikhail Burtsev, and Evgeny Burnaev. Arigraph: learning knowledge graph world models with episodic memory for llm agents. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 2025.

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

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, et al. Lightmem: Lightweight and efficient memory-augmented generation. In *International Conference on Learning Representations*, volume 2026, pp. 98706–98729, 2026.

Pengyu Gao, Jinming Zhao, Xinyue Chen, and Long Yilin. An efficient context-dependent memory framework for llm-centric agents. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pp. 1055–1069, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, Senjie Jin, Jiejun Tan, Yanbin Yin, Jiongnan Liu, Zeyu Zhang, Zhongxiang Sun, Yutao Zhu, Hao Sun, Boci Peng, Zhenrong Cheng, Xuanbo Fan, Jiaxin Guo, Xinlei Yu, Zhenhong Zhou, Zewen Hu, Jiahao Huo, Junhao Wang, Yuwei Niu, Yu Wang, Zhenfei Yin, Xiaobin Hu, Yue Liao, Qiankun Li, Kun Wang, Wangchunshu Zhou, Yixin Liu, Dawei Cheng, Qi Zhang, Tao Gui, Shirui Pan, Yan Zhang, Philip Torr, Zhicheng Dou, Ji-Rong

Wen, Xuanjing Huang, Yu-Gang Jiang, and Shuicheng Yan. Memory in the age of ai agents. *arXiv preprint arXiv:2507.03724*, 2026.

Wei-Chieh Huang, Weizhi Zhang, Yueqing Liang, Yuanchen Bei, Yankai Chen, Tao Feng, xinyu Pan, Zhen Tan, Yu Wang, Tianxin Wei, Shanglin Wu, Ruiyao Xu, Liangwei Yang, Rui Yang, Wooseong Yang, Chin-Yuan Yeh, Hanrong Zhang, Haozhen Zhang, Siqi Zhu, Henry Peng Zou, Wanjia Zhao, Song Wang, Wujiang Xu, Zixuan Ke, Zheng Hui, Dawei Li, Yaozu Wu, Langzhou He, Chen Wang, Xiongxiao Xu, Baixiang Huang, Juntao Tan, Shelby Heinecke, Huan Wang, Caiming Xiong, Ahmed Metwally, Jun Yan, Chen-Yu Lee, Hanqing Zeng, Yinglong Xia, Xiaokai Wei, Ali Payani, Yu Wang, Haitong Ma, Wenya Wang, Chenguang Wang, Yu Zhang, Xin Eric Wang, Yongfeng Zhang, Jiaxuan You, Hanghang Tong, Xiao Luo, Xue Liu, Yizhou Sun, Wei Wang, Julian McAuley, James Zou, Jiawei Han, Philip S. Yu, and Kai Shu. A survey of agent memory in the second half: Towards self-evolving and long-horizon agents. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856.

Shuo Ji, Yibo Li, and Bryan Hooi. Memory is reconstructed, not retrieved: Graph memory for LLM agents. In *Forty-third International Conference on Machine Learning*, 2026.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. Memory OS of AI agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 25961–25970, Suzhou, China, November 2025. Association for Computational Linguistics.

Zhiyu Li, Chenyang Xi, Chunyu Li, Ding Chen, Boyu Chen, Shichao Song, Simin Niu, Hanyu Wang, Jiawei Yang, Chen Tang, et al. Memos: A memory os for ai system. *arXiv preprint arXiv:2507.03724*, 2025.

Junru Lu, Siyu An, Mingbao Lin, Gabriele Pergola, Yulan He, Di Yin, Xing Sun, and Yunsheng Wu. Memochat: Tuning llms to use memos for consistent long-range open-domain conversation. *arXiv preprint arXiv:2308.08239*, 2023.

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

Andy Nguyen, Danh Doan, Hoang Pham, Bao Ha, Dat Pham, Linh Nguyen, Hieu Nguyen, Thien Nguyen, Cuong Do, Phat Nguyen, et al. Byterover: Agent-native memory through llm-curated hierarchical context. *arXiv preprint arXiv:2604.01599*, 2026.

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.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, et al. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 963–981, 2024.

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.

Alireza Rezazadeh, Zichao Li, Wei Wei, and Yujia Bao. From isolated conversations to hierarchical schemas: Dynamic tree memory representation for llms. In *International Conference on Learning Representations*, volume 2025, pp. 990–1023, 2025.

Rana Salama, Jason Cai, Michelle Yuan, Anna Currey, Monica Sunkara, Yi Zhang, and Yassine Benajiba. MemInsight: Autonomous memory augmentation for LLM agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33136–33152, Suzhou, China, November 2025. Association for Computational Linguistics.

Alina Shutova, Alexandra Olenina, Ivan Vinogradov, and Anton Sinitsin. Evaluating memory structure in llm agents. *arXiv preprint arXiv:2507.03724*, 2026.

Miao Su, Yucan Guo, Zhongni Hou, Long Bai, Zixuan Li, Yufei Zhang, Guojun Yin, Wei Lin, Xiaolong Jin, Jiafeng Guo, and Xueqi Cheng. Beyond dialogue time: Temporal semantic memory for personalized LLM agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 29935–29951, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Haoran Sun, Shaoning Zeng, and Bob Zhang. H-mem: Hierarchical memory for high-efficiency long-term reasoning in llm agents. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 341–350, 2026.

Kaixiang Wang, Yidan Lin, Zihan Wang, Bunyod Suvonov, Zhaojiacheng Zhou, Yuxiang Zheng, Jiaxi Cao, Zhiheng Dong, Chentao Wu, Jiong Lou, and Jie LI. E-mem: Multi-agent based episodic context reconstruction for LLM agent memory. In *Forty-third International Conference on Machine Learning*, 2026.

Tianxin Wei, Noveen Sachdeva, Benjamin Coleman, Zhankui He, Yuanchen Bei, Xuying Ning, Mengting Ai, Yunzhe Li, Jingrui He, Ed H Chi, et al. Evo-memory: Benchmarking llm agent test-time learning with self-evolving memory. *arXiv preprint arXiv:2511.20857*, 2025.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 86809–86836, 2025.

Yanchen Wu, Tenghui Lin, Yingli Zhou, Fangyuan Zhang, Qintian Guo, Xun Zhou, Sibo Wang, Xilin Liu, Yuchi Ma, and Yixiang Fang. Memory in the llm era: Modular architectures and strategies in a unified framework. *arXiv preprint arXiv:2507.03724*, 2026a.

Zhaofen Wu, Hanrong Zhang, Fulin Lin, Wujiang Xu, Xinran Xu, Yankai Chen, Henry Peng Zou, Shaowen Chen, Weizhi Zhang, Xue Liu, et al. Gam: Hierarchical graph-based agentic memory for llm agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 34647–34664, 2026b.

Siyu Xia, Zekun Xu, Jiajun Chai, Wentian Fan, Yan Song, Xiaohan Wang, Guojun Yin, Wei Lin, Haifeng Zhang, and Jun Wang. From experience to strategy: Empowering llm agents with trainable graph memory. *arXiv preprint arXiv:2507.03724*, 2025.

Buqiang Xu, Yijun Chen, Jizhan Fang, Ruobin Zhong, Yunzhi Yao, Yuqi Zhu, Lun Du, and Shumin Deng. Structmem: Structured memory for long-horizon behavior in llms. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 122–146, 2026a.

Derong Xu, Yi Wen, Pengyue Jia, Yingyi Zhang, Wenlin Zhang, Yichao Wang, Huifeng Guo, Ruiming Tang, Xiangyu Zhao, Enhong Chen, and Tong Xu. From single to multi-granularity: Toward long-term memory association and selection of conversational agents. In *International Conference on Learning Representations*, volume 2026, pp. 91581–91617, 2026b.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *Advances in Neural Information Processing Systems*, 38:17577–17604, 2026c.

BY Yan, Chaofan Li, Hongjin Qian, Shuqi Lu, and Zheng Liu. General agentic memory via deep research. *arXiv preprint arXiv:2511.18423*, 2025.

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

Ke Yang, Zixi Chen, Xuan He, Jize Jiang, Michel Galley, Chenglong Wang, Jianfeng Gao, Jiawei Han, and ChengXiang Zhai. Plugmem: A task-agnostic plugin memory module for LLM agents. In *Forty-third International Conference on Machine Learning*, 2026.

Wei Yang, Jinwei Xiao, Hongming Zhang, Qingyang Zhang, Yanna Wang, and Bo Xu. Coarse-to-fine grounded memory for LLM agent planning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13029–13056, Suzhou, China, November 2025b. Association for Computational Linguistics.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, et al. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. In *International Conference on Learning Representations*, volume 2026, pp. 39458–39486, 2026.

Guibin Zhang, Muxin Fu, Kun Wang, Frank Wan, Miao Yu, and Shuicheng Yan. G-memory: Tracing hierarchical memory for multi-agent systems. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 12988–13018. Curran Associates, Inc., 2025a. doi: 10.52202/085713-0439.

Haozhen Zhang, Haodong Yue, Tao Feng, Quanyu Long, Jianzhu Bao, Bowen Jin, Weizhi Zhang, Xiao Li, Jiaxuan You, Chengwei Qin, et al. Learning query-aware budget-tier routing for runtime agent memory. *arXiv preprint arXiv:2602.06025*, 2026a.

Jiaquan Zhang, Chaoning Zhang, Shuxu Chen, Zhenzhen Huang, Pengcheng Zheng, Zhicheng Wang, Ping Guo, Fan Mo, Sung-Ho Bae, Jie Zou, et al. Lightweight llm agent memory with small language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12914–12929, 2026b.

Ningning Zhang, Xingxing Yang, Zhizhong Tan, Weiping Deng, and Wenyong Wang. Himem: Hierarchical long-term memory for llm long-horizon agents. *arXiv preprint arXiv:2601.06377*, 2026c.

Qizheng Zhang, Michael Wornow, and Kunle Olukotun. Agentic plan caching: Test-time memory for fast and cost-efficient llm agents. *Advances in Neural Information Processing Systems*, 38: 103270–103296, 2026d.

Xiaohui Zhang, Zequn Sun, Chengyuan Yang, Yaqin Jin, Yazhong Zhang, and Wei Hu. Actmem: Bridging the gap between memory retrieval and reasoning in llm agents. *arXiv preprint arXiv:2603.00026*, 2026e.

Yujie Zhang, Weikang Yuan, and Zhuoren Jiang. Bridging intuitive associations and deliberate recall: Empowering llm personal assistant with graph-structured long-term memory. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 17533–17547, 2025b.

Zeyu Zhang, Quanyu Dai, Rui Li, Xiaohe Bo, Xu Chen, and Zhenhua Dong. Learn to memorize: Optimizing llm-based agents with adaptive memory framework. *arXiv preprint arXiv:2507.03724*, 2025c.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 19724–19731, 2024.

Chenyu Zhou, Huacan Chai, Wenteng Chen, Zihan Guo, Rong Shan, Yuanyi Song, Tianyi Xu, Yingxuan Yang, Aofan Yu, Weiming Zhang, Congming Zheng, Jiachen Zhu, Zeyu Zheng, Zhuosheng Zhang, Xingyu Lou, Changwang Zhang, Zhihui Fu, Jun Wang, Weiwen Liu, Jianghao Lin, and Weinan Zhang. Externalization in llm agents: A unified review of memory, skills, protocols and harness engineering. *arXiv preprint arXiv:2507.03724*, 2026.

Yuqi Zhu, Shuofei Qiao, Yixin Ou, Shumin Deng, Shiwei Lyu, Yue Shen, Lei Liang, Jinjie Gu, Huajun Chen, and Ningyu Zhang. Knowagent: Knowledge-augmented planning for llm-based agents. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3709–3732, 2025.

A Appendix

A.1 Related Work

A.1.1 Agent Memory Management

Recent surveys characterize agent memory as a system-level process involving memory formation, organization, retrieval, and evolution, rather than a passive store of dialogue history (Hu et al., 2026; Huang et al., 2026; Wu et al., 2026a; Zhou et al., 2026). Existing methods can be broadly grouped into flat, hierarchical, and structured memory according to how historical information is represented and maintained.

Flat-memory systems store relatively independent summaries, facts, or attributed entries. MemoryBank maintains long-term conversational records with an Ebbinghaus-inspired forgetting mechanism (Zhong et al., 2024), while MemoChat organizes dialogue into topic-level memos and MemInsight enriches individual memories with automatically generated semantic and contextual attributes (Lu et al., 2023; Salama et al., 2025). Mem0 is a representative production-oriented framework that extracts salient facts from interactions, consolidates or updates conflicting information, and stores the resulting memories in a scalable vector database; its graph-enhanced variant additionally models entity relations (Chhikara et al., 2025). Zhang et al. (2026d) stores reusable plan templates instead of raw interaction histories, whereas Yu et al. (2026) continually compresses long inputs into a bounded recurrent memory state. These designs prioritize efficient memory maintenance and generally leave the discovery of relations among dispersed memories to retrieval or downstream reasoning.

Hierarchical methods organize information across storage tiers or abstraction levels. Packer et al. (2023) manages the context window and external storage as a virtual-memory hierarchy, while Kang et al. (2025) and Li et al. (2025) introduce staged personal-memory layers or schedulable heterogeneous memory units. H-MEM, MemTree, and HiMem build semantic abstraction hierarchies, dynamic trees, or episode-to-note mappings to support coarse-to-fine access (Sun et al., 2026; Rezazadeh et al., 2025; Zhang et al., 2026c). LightMem adopts a lightweight three-stage design consisting of sensory memory, topic-aware short-term memory, and offline-updated long-term memory, thereby moving expensive consolidation away from online interaction (Fang et al., 2026). Zhang et al. (2026b) delegates short-, mid-, and long-term memory operations to specialized small language models and combines offline consolidation with intent-conditioned access. Nguyen et al. (2026) and Wang et al. (2026) similarly use hierarchical context organization or distributed episodic contexts, while Yang et al. (2025b) and Zhang et al. (2026a) organize experiences by granularity or access budget (Nguyen et al., 2026; Wang et al., 2026; Yang et al., 2025b; Zhang et al., 2026a). These methods emphasize scalable storage, compression, and efficient routing across memory levels.

Structured memory makes cross-memory relations explicit. A-MEM is a representative agentic-memory method that converts experiences into attributed notes, links them through a Zettelkasten-style network, and evolves related memories as new information arrives (Xu et al., 2026c). AriGraph, Zep, Associa, and CDMem use world-model, bi-temporal, event-centric, or context-dependent graphs to connect entities and experiences (Anokhin et al., 2025; Rasmussen et al., 2025; Zhang et al., 2025b; Gao et al., 2025). Zhang et al. (2025a) extends hierarchical graphs to multi-agent collaboration, and Wu et al. (2026b) separates rapidly changing event progression from a relatively stable topic-associative network. ActMem constructs a causal-semantic memory graph, completes implicit constraints, and uses the graph to support causal and counterfactual reasoning rather than treating retrieval as an isolated lookup step (Zhang et al., 2026e). MRAgent likewise treats memory access as active reconstruction over Cue–Tag–Content paths (Ji et al., 2026). Other structured systems emphasize provenance, event binding, temporal continuity, executable knowledge, or strategic abstraction (Yang et al., 2026; Xu et al., 2026a; Su et al., 2026; Zhu et al., 2025; Xia et al., 2025). Collectively, these methods demonstrate that explicit structure can move memory access beyond isolated similarity matching by connecting dispersed evidence. Building on this direction, we organize long-term memories from the complementary perspectives of logical association and temporal evolution, enabling weakly matched evidence to be recovered through its dependencies with retrieved memories.

A.1.2 Agent Memory Retrieval

Similarity-based retrieval ranks memories according to semantic proximity or attributed relevance. MemoryBank, Mem0, MemoChat, A-MEM, MemInsight, LightMem, and Lightweight LLM Agent Memory all follow this general pattern despite substantial differences in memory construction (Zhong et al., 2024; Chhikara et al., 2025; Lu et al., 2023; Xu et al., 2026c; Salama et al., 2025; Fang et al., 2026; Zhang et al., 2026b). In particular, Mem0 performs vector retrieval over consolidated facts, while A-MEM supplements semantic relevance with the attributes and links of its evolving note network. Fang et al. (2026) retrieves from offline-consolidated memories; Zhang et al. (2026b) further rewrites requests into intent-conditioned hypothetical queries, allocates a retrieval budget, and reranks candidates under metadata constraints. Similarity-based retrieval is efficient, but semantic proximity does not always reflect reasoning relevance: useful memories with weak query similarity may be overlooked, while highly similar but less informative memories can receive high priority.

Adaptive retrieval dynamically determines which memory level or module should be accessed and how much context should be returned. MemGAS directly addresses the mismatch between fixed memory granularity and diverse queries: it associates memories at multiple granularities through Gaussian-mixture clustering, uses an entropy-based router to select the appropriate granularity, and applies LLM filtering before generation (Xu et al., 2026b). Related approaches route memory modules by query-dependent cost, learn retrieval and aggregation policies from interaction feedback, or activate distributed episodic contexts for local reasoning (Zhang et al., 2026a; 2025c; Wang et al., 2026). Yang et al. (2025b) retrieves experiences and actionable tips and invokes fine-grained grounding when anomalies arise. By adapting granularity, access budget, or filtering strategy to the current query, these methods partially alleviate the limitations of fixed similarity retrieval and reduce irrelevant context. Our focus is complementary: rather than primarily deciding where or how much to retrieve, we exploit dependencies among memories to recover weakly matched evidence and estimate each candidate's information contribution during reranking.

Structure-aware retrieval provides another way to reduce dependence on direct query similarity. AriGraph combines episodic similarity with semantic-graph access; Associa extracts evidence-rich subgraphs; Zep combines vector and full-text search with graph traversal; and PlugMem interleaves similarity search, concept-level expansion, and provenance tracing (Anokhin et al., 2025; Zhang et al., 2025b; Rasmussen et al., 2025; Yang et al., 2026). CDMem, G-Memory, and graph-based GAM retrieve through contextual or hierarchical graph indices, with GAM jointly using event progression and topic association to improve recall (Gao et al., 2025; Zhang et al., 2025a; Wu et al., 2026b). Zhang et al. (2026e) further couples graph retrieval with causal, commonsense, and counterfactual reasoning, while Ji et al. (2026) iteratively explores and prunes associative paths as evidence accumulates. Su et al. (2026) adds semantic-time parsing, temporal filtering, and temporal-knowledge-graph evidence to obtain temporally valid memories. KnowAgent and trainable graph memory retrieve executable action paths or reward-weighted strategic abstractions for planning (Zhu et al., 2025; Xia et al., 2025). These methods demonstrate that structural relations can uncover evidence beyond direct semantic matching. CoRelMem follows this general direction but explicitly separates cross-memory logical association from temporal state evolution and uses both structures as complementary retrieval paths.

Runtime reconstruction treats retrieval as the process of building a query-specific working context rather than selecting a fixed list of stored items. Yan et al. (2025) retains complete interaction histories in a universal page store and conducts query-directed deep research to construct a just-in-time context, guided by lightweight offline memory. Li et al. (2025) similarly schedules task-specific indices, temporal chains, and entity relations across heterogeneous memory types, whereas Yu et al. (2026) eliminates explicit retrieval by learning what state to preserve during sequential reading. Evo-Memory and StructMemEval further argue that memory systems should be evaluated not only by fact recall, but also by continual adaptation and the ability to discover and maintain task-appropriate structures (Wei et al., 2025; Shutova et al., 2026). These approaches broaden retrieval from one-shot matching to context construction and adaptive memory use.

Overall, memory retrieval has progressively moved beyond simple semantic matching through adaptive routing, structural expansion, temporal constraints, runtime reconstruction, and learned filtering. These advances alleviate the limitations of similarity-based retrieval from different perspectives. Nevertheless, two challenges remain central to our study: relevant evidence may exhibit weak

semantic similarity to the query, while highly similar memories may contribute little useful information. CoRelMem addresses these issues from a complementary perspective by using cross-memory association and temporal evolution to expand weakly matched evidence, and conditional information gain as an additional reranking signal to distinguish semantic relevance from actual information contribution.

A.2 Theoretical Analysis of CoRelMem

This section further explains the design of CoRelMem from a theoretical perspective. We first analyze how Coherent Relation Induction expands the reachable range of relevant evidence through cross-memory structures. We then explain the modeling foundations of the memory association graph and temporal trajectories in CRI from the perspectives of local conditional dependency and local state transition. Finally, we analyze Information-Aware Reranking from an information-theoretic perspective.

A.2.1 Expansion of Relevant Evidence Reachability by CRI

Let $\mathcal{R}_q$ denote the set of all relevant memories corresponding to query q, and let $\mathcal{S}_q$ denote the seed memory set obtained by direct retrieval. If retrieval relies only on direct query–memory matching, the retrievable relevant evidence is limited to $\mathcal{R}_q \cap \mathcal{S}_q$. However, in long-term interaction scenarios, some relevant evidence may have weak direct semantic matching with the query itself, while maintaining logical associations or temporal connections with already retrieved memories. CRI explicitly organizes these two types of relations through the memory association graph $\mathcal{G}^A$ and the temporal trajectory set $\mathcal{T}$.

Let $\mathcal{N}_A(\mathcal{S}_q)$ denote the candidate set expanded from the seed set along the memory association graph, and let $\mathcal{N}_T(\mathcal{S}_q)$ denote the candidate set further obtained from the temporal trajectories containing the seeds. The candidate set after dual-path expansion can then be expressed as

$$\mathcal{C}_q = \mathcal{S}_q \cup \mathcal{N}_A(\mathcal{S}_q) \cup \mathcal{N}_T(\mathcal{S}_q).$$

Since

$$\mathcal{S}_q \subseteq \mathcal{C}_q,$$

at the candidate level without applying a fixed Top-K truncation, we have

$$|\mathcal{R}_q \cap \mathcal{C}_q| \geq |\mathcal{R}_q \cap \mathcal{S}_q|.$$

That is,

$$\mathrm{Recall}(\mathcal{C}_q) \geq \mathrm{Recall}(\mathcal{S}_q).$$

Furthermore, if there exists at least one relevant memory that is not directly retrieved but can be reached from the seed memories through the structures constructed by CRI, namely,

$$m \in \mathcal{R}_q \setminus \mathcal{S}_q$$

and

$$m \in \mathcal{N}_A(\mathcal{S}_q) \cup \mathcal{N}_T(\mathcal{S}_q),$$

then

$$\mathrm{Recall}(\mathcal{C}_q) > \mathrm{Recall}(\mathcal{S}_q).$$

Therefore, CRI does not replace direct semantic retrieval, but instead expands the reachable space of relevant evidence on top of the existing seed memories. When relevant evidence is distributed across memories, exhibits logical dependencies, or follows temporal continuity, such structural expansion enables memories that are difficult to retrieve through direct semantic matching to re-enter the candidate set.

A.2.2 Modeling Interpretation of CRI

CRI organizes long-term memories into two complementary structures. The memory association graph describes horizontal dependencies across memories, while temporal trajectories describe the longitudinal evolution of event states. These two structures can be understood from the perspectives of MRF-like local conditional dependency and Markov-like local state transition, respectively.

(1) MRF-like Interpretation of the Memory Association Graph For two candidate memories m_i and m_j, let C_{ij} denote their local context. If, given C_{ij}, additionally observing m_i changes the conditional distribution of m_j, namely,

$$P(m_j \mid C_{ij}, m_i) \neq P(m_j \mid C_{ij}),$$

then m_i and m_j are not conditionally independent under this local context. This perspective is similar to the idea of describing variable associations through local conditional dependencies in a Markov Random Field. Therefore, the memory association graph can be viewed as an MRF-like local dependency structure. Its purpose, however, is not to explicitly learn a probabilistic MRF, but to use local information dependency to characterize which direct relations between memories are worth preserving. CoRelMem uses Conditional Likelihood Gain (Eqn. 2) to approximately measure such dependency, namely $\Delta(m_j; m_i \mid C_{ij})$. If the conditional predictability of m_j improves significantly after incorporating m_i, this indicates that m_i provides additional information for m_j. The aggregation of bidirectional CLG is further used to characterize the overall information dependency strength between the two memories.

(2) Markov-like Interpretation of Temporal Trajectories For a temporal memory sequence within the same continuously evolving event thread,

$$m_1^T, m_2^T, \ldots, m_L^T,$$

CoRelMem adopts an approximate first-order local state transition assumption, under which the subsequent state mainly depends on the current state:

$$P(m_{t+1}^T \mid m_1^T, \ldots, m_t^T) \approx P(m_{t+1}^T \mid m_t^T).$$

This assumption corresponds to the local transition idea of a first-order Markov chain. Its purpose is not to assume that long-term events strictly satisfy the Markov property, but to use local state dependencies to reduce the difficulty of modeling complex temporal relations. Therefore, temporal trajectories do not simply connect neighboring events according to timestamps. Instead, candidate events are first required to belong to the same continuously evolving thread and form reasonable preceding and succeeding states. On this basis, $\Delta(m_{t+1}^T; m_t^T)$ is used to measure the predictive contribution of the current event to the next state, and local transition probabilities are further used to select a more reasonable successor.

From this perspective, CRI jointly models two complementary types of relations, constructing the association graph from the conditional dependency perspective and temporal trajectories from the state transition perspective. It should be emphasized that both MRF-like and Markov-like are modeling interpretations rather than strict probabilistic graphical models. CoRelMem does not explicitly learn MRF parameters, nor does it assume that all temporal processes strictly satisfy the first-order Markov property.

A.2.3 Information-Theoretic Interpretation of IAR

After structural expansion, candidate memories already have a base evidence score $S_{\text{evi}}(q, U)$, which reflects the relevance between candidate U and the current query q in the original retrieval process. However, relying solely on this score may still make it difficult to distinguish superficially relevant candidates with limited information contribution. Therefore, Information-Aware Reranking introduces an information signal based on conditional likelihood gain while preserving the original evidence relevance.

For query q and candidate evidence U, the information gain term is defined as $\Delta(q; U)$, i.e.,

$$S_{\text{info}}(q, U) = \Delta(q; U) = \ell(q \mid U) - \ell(q)$$

where

$$\ell(q \mid U) = \frac{1}{|q|} \log P_\theta(q \mid U).$$

If length normalization is temporarily ignored, we have

$$\log P(q \mid U) - \log P(q) = \log \frac{P(q \mid U)}{P(q)}.$$

According to Bayes' rule, this can be further written as

$$\log \frac{P(q,U)}{P(q)P(U)},$$

which has the same form as Pointwise Mutual Information:

$$\mathrm{PMI}(q,U) = \log \frac{P(q,U)}{P(q)P(U)}.$$

Therefore, $S_{\mathrm{info}}(q,U)$ can be understood as a length-normalized pointwise information gain, which measures the extent to which the conditional predictability of query q improves after observing candidate memory U. This signal mainly represents the conditional information dependency between the candidate and the query.

Furthermore, from the perspective of mutual information,

$$I(Q;U) = H(Q) - H(Q \mid U),$$

which represents the average reduction in the uncertainty of random variable Q after observing U. IAR does not directly estimate the overall mutual information, but instead approximates such sample-level information contribution for a specific query-memory pair through the conditional likelihood of the language model. Finally, IAR combines the original evidence score with the information gain signal:

$$S_{\mathrm{IAR}}(q,U) = (1-\beta)S_{\mathrm{evi}}(q,U) + \beta S_{\mathrm{info}}(q,U),$$

where β controls the trade-off between retrieval relevance and information contribution. Accordingly, IAR introduces a complementary information-theoretic signal while preserving the original evidence relevance, enabling more fine-grained prioritization of structurally expanded candidates.

Similar information signals based on language-model likelihood differences have also been used in long-term memory research. For example, ActMem (Zhang et al., 2026e) uses a PMI-like likelihood difference to verify candidate memory relations and reduce spurious relations generated by LLMs. In contrast, CoRelMem applies this information-theoretic idea at two different stages. During memory organization, CLG serves as a unified information dependency measure to assist the induction of associative relations and temporal transitions; during retrieval, IAR combines conditional information gain with the existing evidence score to rerank candidates after structural expansion.

A.3 Additional Method Details

This section provides additional implementation details omitted from the main text, including memory-type definitions, successor selection in relation induction, candidate construction in dual-path retrieval, and the operational details of IAR.

A.3.1 Structured Memory Extraction

The extracted atomic memories are categorized into three types according to their roles in long-term memory. *General Fact* represents relatively stable information that is not strongly tied to a specific time point, such as personal attributes, relationships, long-term states, and stable preferences. *Temporal Event* describes events with explicit temporal attributes or state changes, such as specific experiences, actions, or stage transitions. *Feedback* mainly contains short-lived interaction responses, including greetings, acknowledgments, and temporary feedback, which usually provide limited long-term memory value. Only General Facts and Temporal Events are retained for subsequent structural organization, while Feedback memories are excluded from relation construction.

A.3.2 Memory Organization via Coherent Relation Induction

(1) Memory Association Graph Construction The two directional dependencies between a candidate memory pair are computed separately because the information contribution from m_i to m_j is not necessarily identical to that from m_j to m_i. Their bidirectional CLG scores are aggregated by simple averaging:

$$S^A_{ij} = \frac{g_{i\to j} + g_{j\to i}}{2}. \tag{11}$$

After LLM-based relation judgment and CLG-based filtering, the retained association strength is used as the edge weight of the memory association graph during query-time propagation. For an association edge (m_i, m_j) with weight w_{ij}, its normalized association confidence is defined as

$$W_{ij} = \frac{w_{ij}}{\sum_{m_k \in \mathcal{N}(m_i)} w_{ik}}, \tag{12}$$

where $\mathcal{N}(m_i)$ denotes the neighbors of m_i. This normalization preserves the relative strength of outgoing associations, allowing query-time propagation to favor more strongly connected memories rather than treating all association edges equally.

(2) Temporal Trajectory Construction For a temporal memory m_i^T, only successor candidates that pass the two-stage LLM judgment and satisfy $g_{i\to j}^T > 0$ are retained. Let $\mathcal{C}^T(i)$ denote the valid successor set. When multiple valid successors remain, their temporal transition gains are normalized as

$$P_{ij}^T = \frac{\exp\left(g_{i\to j}^T/\tau_T\right)}{\sum_{k\in\mathcal{C}^T(i)} \exp\left(g_{i\to k}^T/\tau_T\right)}, \tag{13}$$

where τ_T is a temperature parameter. The local successor is then selected by

$$j^* = \arg\max_{j\in\mathcal{C}^T(i)} P_{ij}^T. \tag{14}$$

This selection favors coherent state evolution rather than simply linking temporally adjacent events.

A.3.3 Knowledge-Guided Dual-Path Memory Retrieval

At query time, the top $K_D = 60$ results ranked by the semantic–structural score defined in the main text are retained as base candidates. A base candidate may correspond to either a single atomic memory or a memory bundle induced by a multi-hop knowledge path. These candidates are then expanded into atomic memories, from which the top $K_S = 10$ memories are selected as the seed set $\mathcal{S}_q$ for structural expansion.

(1) Association-based Retrieval Starting from the seed set $\mathcal{S}_q$, retrieval is restricted to a local subgraph of the memory association graph:

$$V_q^A = \left\{ m \mid \min_{s\in\mathcal{S}_q} d_{\mathcal{G}^A}(s, m) \le H_A \right\}, \tag{15}$$

where $d_{\mathcal{G}^A}$ denotes the shortest-path distance in the association graph. Before graph propagation, local nodes are filtered according to their query–memory semantic relevance. The personalized restart distribution used by PPR is computed from the semantic relevance of the seeds:

$$v_i = \frac{\exp\left(S_{\text{sem}}(q, s_i)/T\right)}{\sum_{s_j\in\mathcal{S}_q} \exp\left(S_{\text{sem}}(q, s_j)/T\right)}, \tag{16}$$

After PPR, CoRelMem recovers the complete association path from a seed to each selected target memory. For a path $\pi = (m_0, m_1, \ldots, m_L)$, its average association strength is defined as

$$S_A(\pi) = \frac{1}{L}\sum_{l=0}^{L-1} w_{m_l, m_{l+1}}. \tag{17}$$

When multiple paths connect a seed to the same target, the path with the highest average association strength is preferred, with shorter paths favored in case of ties. The recovered path is treated as a single association evidence unit.

(2) Temporal Trajectory Retrieval If a seed memory belongs to a temporal trajectory, CoRelMem searches for a contiguous segment containing that seed. For a trajectory $T_p = (m_1^T, \ldots, m_L^T)$, a candidate temporal segment is written as

$$U_T = (m_a^T, m_{a+1}^T, \ldots, m_b^T), \tag{18}$$

subject to

$$b - a + 1 \leq L_T, \qquad U_T \cap \mathcal{S}_q \neq \varnothing, \tag{19}$$

When a trajectory contains multiple seeds, the shortest contiguous segment covering them is preferred. If the segment exceeds the maximum length, length-constrained windows containing seed memories are considered. The highest-scoring segment under the temporal relevance function defined in the main text is retained as one temporal evidence unit. After dual-path expansion, the candidate pool contains direct, association, and temporal evidence units.

A.3.4 Information-Aware Reranking

(1) Structured Unit Filtering Structural expansion may produce multi-memory units whose combination does not provide additional information beyond their strongest component. For a structured evidence unit U, let

$$\Phi(U) = \{m_1, m_2, \ldots, m_{|U|}\} \tag{20}$$

denote its constituent atomic memories. We define the structural CLG gain as

$$G(U) = S_{\mathrm{CLG}}(q, U) - \max_{m_i \in \Phi(U)} S_{\mathrm{CLG}}(q, m_i), \tag{21}$$

where $S_{\mathrm{CLG}}(q, U) = \Delta(q; U)$. Only association and temporal units satisfying $G(U) > \tau_G$ are retained, with $\tau_G = 0$ by default. Direct evidence units are not subject to this filtering. This criterion requires a structured unit to provide greater conditional information than any of its individual memories.

(2) Information-aware Fusion Final reranking is performed at the evidence-unit level. For each candidate unit U, semantic relevance is computed from the complete unit text:

$$S_D(q, U) = \cos\left(E(q), E(U)\right), \tag{22}$$

where $E(\cdot)$ denotes the text encoder. The information contribution of U is measured using CLG:

$$S_{\mathrm{CLG}}(q, U) = \overline{\log P_\theta(q \mid U)} - \overline{\log P_\theta(q)}, \tag{23}$$

where the conditional log-likelihood is averaged over query tokens:

$$\overline{\log P_\theta(q \mid U)} = \frac{1}{|q|} \sum_{t=1}^{|q|} \log P_\theta\left(q_t \mid U, q_{<t}\right). \tag{24}$$

Since semantic similarity and CLG lie on different numerical scales, both scores are transformed into within-query percentile ranks:

$$R_D(U) = \mathrm{Percentile}\left(S_D(q, U)\right), \tag{25}$$

$$R_C(U) = \mathrm{Percentile}\left(S_{\mathrm{CLG}}(q, U)\right), \tag{26}$$

where $R_D(U), R_C(U) \in [0, 1]$. The operational IAR score is then

$$S_{\mathrm{IAR}}(q, U) = (1 - \beta) R_D(U) + \beta R_C(U). \tag{27}$$

The PPR score and temporal segment score are used to discover and construct structured candidates, but are not directly included in the final IAR fusion.

(3) Disjoint Top-K Selection To avoid repeatedly presenting the same atomic memory through different evidence units, the final retrieval stage adopts a disjoint selection strategy. A candidate U can be selected only if

$$\Phi(U) \cap \left(\bigcup_{V \in \mathcal{R}_q} \Phi(V) \right) = \varnothing, \tag{28}$$

where $\mathcal{R}_q$ denotes the set of already selected evidence units. Candidates are scanned in descending order of S_{IAR} until the retrieval budget K_R is reached.

Table 6: Statistics of the datasets.

Dataset	# Instances	# Questions	Avg. Sessions	Avg. Turns	Avg. Tokens
LoCoMo	10	1,986	27.20	588.20	13.4K
LongMemEval-S	500	500	47.73	493.50	78.4K

A.4 Detailed Experimental Setup

A.4.1 Data Statistics

We conduct experiments on two long-term memory benchmarks, **LoCoMo** and **LongMemEval-S**, to evaluate the model's capabilities in memory retrieval and cross-memory information integration over long interaction histories. Both datasets are based on multi-session histories, but they differ in dialogue scale, context length, and question types. LoCoMo consists of a small number of extremely long dialogue instances, while LongMemEval-S constructs a long interaction history for each test question. Table 6 summarizes the basic statistics of the two datasets. Here, Avg. Sessions, Avg. Turns, and Avg. Tokens denote the average number of sessions, dialogue turns, and text tokens per instance, respectively. As shown in the table, each LoCoMo instance contains more continuous dialogue turns, whereas LongMemEval-S contains more independent sessions and substantially longer contexts. These differences in dataset scale and history organization allow the two benchmarks to evaluate long-term memory systems from different perspectives in terms of organizing, retrieving, and reasoning over long interaction histories.

LoCoMo LoCoMo consists of 10 long-term dialogue instances, where each instance contains multiple sessions that unfold over time. The full dataset contains 1,986 questions. As shown in Table 7, the questions in LoCoMo are divided into five categories: multi-hop reasoning, temporal reasoning, open-domain, single-hop reasoning, and adversarial questions. Since this work mainly focuses on the retrieval and reasoning capabilities of long-term memory systems on answerable questions, we use the first four categories in our experiments, resulting in 1,540 samples. The remaining 446 adversarial questions are excluded from the main performance evaluation.

Table 7: Distribution of question types in the datasets.

Dataset	Question Type	# Questions
LoCoMo	Multi-hop	282
	Temporal	321
	Open-domain	96
	Single-hop	841
	Adversarial	446
LongMemEval-S	Single-session User	70
	Single-session Assistant	56
	Single-session Preference	30
	Multi-session	133
	Knowledge Update	78
	Temporal Reasoning	133

LongMemEval-S LongMemEval-S is a subset of LongMemEval and contains 500 test instances, where each instance corresponds to one question and its associated long interaction history. Compared with LoCoMo, each LongMemEval-S instance has a longer context and consists of more independent sessions. As summarized in Table 7, LongMemEval-S divides the questions into six categories: single-session user information, single-session assistant information, single-session preference, multi-session reasoning, knowledge update, and temporal reasoning.

A.4.2 Metrics

We evaluate long-term memory systems from two aspects: generation quality and retrieval quality. Generation is evaluated by answer correctness and reference-based similarity, while retrieval is evaluated by relevant evidence recall and ranking quality.

For generation evaluation, we assess the final generated answers from two aspects: answer correctness and reference-based similarity. Answer correctness measures whether the model can generate the correct answer based on the retrieved memories, while reference-based similarity evaluates the consistency between the generated and reference answers in terms of lexical matching and textual

overlap. Specifically, we use Accuracy (ACC), F1, BLEU-1, ROUGE-1, ROUGE-2, and ROUGE-L as evaluation metrics. ACC measures the correctness of the final generated answers. For each test question, an LLM judges whether the generated answer is correct, and ACC is computed as the proportion of correctly answered samples. F1 measures the token-level overlap between the generated and reference answers. Precision and Recall are first computed based on the number of overlapping tokens, and their harmonic mean is used as the F1 score. BLEU-1 measures unigram-level matching between the generated and reference answers, mainly reflecting exact lexical overlap. ROUGE-1 measures unigram-level content overlap between the generated and reference answers. ROUGE-2 measures bigram-level content overlap, further capturing local word-order and phrase-level matching. ROUGE-L measures the overall sequence similarity between the generated and reference answers based on the Longest Common Subsequence (LCS), reflecting consistency in content order and overall expression.

For retrieval evaluation, we use Recall@K, MRR@K, and NDCG@K to evaluate the recall of relevant evidence and the quality of the retrieved ranking. Recall@K measures the proportion of ground-truth relevant evidence covered by the top-K retrieved results. MRR@K measures the ranking position of the first relevant evidence item. If no relevant evidence appears within the top-K results, its reciprocal rank is set to 0. NDCG@K measures the overall ranking quality of the top-K retrieved results.

A.4.3 Baselines

To comprehensively compare CoRelMem with existing long-term memory methods, we select Mem0, A-Mem, LightMem, and MemGAS as the main baselines. These methods cover different technical paradigms, ranging from similarity-driven memory retrieval and structured memory organization to hierarchical memory management and adaptive retrieval, enabling the evaluation of CoRelMem from different perspectives in long-term memory organization and retrieval.

- **Mem0** (Chhikara et al., 2025) is a general-purpose memory framework for long-term interactions. It extracts key information from historical interactions and stores it as independent memory units, which are later retrieved according to their semantic relevance to the current query. Its relatively simple pipeline represents long-term memory methods centered on independent memory units and similarity-based retrieval.
- **A-Mem** (Xu et al., 2026c) emphasizes structured organization of long-term memory by establishing associations among memory units, thereby enhancing information connectivity and subsequent retrieval. Compared with conventional independent memory storage, it further considers structural relations among memories and serves as a representative baseline for structured long-term memory methods.
- **LightMem** (Fang et al., 2026) focuses on improving the storage and retrieval efficiency of long interaction histories. It organizes memory in a lightweight and hierarchical manner, reducing redundant information while preserving content useful for downstream tasks. Its key characteristic is the use of memory representations at different granularities for efficient long-term information management, making it representative of hierarchical memory organization and efficient retrieval methods.
- **MemGAS** (Xu et al., 2026b) addresses the limitations of fixed retrieval strategies in complex long-term memory tasks. It dynamically adjusts the retrieval process according to the current query and adaptively selects and combines relevant memories to support multi-evidence retrieval and reasoning for complex questions.

A.4.4 Implementation Details

We use Qwen3-4B (Yang et al., 2025a) and Llama-3.2-3B (Grattafiori et al., 2024) as the primary LLM backbones to evaluate the effectiveness of CoRelMem across different model backbones. Our implementation is mainly developed based on the open-source codebase of LightMem (Fang et al., 2026). For the baselines, LightMem follows its original implementation, while Mem0 (Chhikara et al., 2025) and A-Mem (Xu et al., 2026c) are run using the implementations integrated into the LightMem repository. MemGAS (Xu et al., 2026b) is implemented using its official source code. During memory construction, we employ LLMLingua-2 (Pan et al., 2024) to compress the origi-

nal conversations with a compression rate of 0.6. For retrieval, all-MiniLM-L6-v2 is used as the embedding model, and cosine similarity is adopted to measure semantic relevance.

By default, the association graph threshold is set to $\tau_A = 0.6$, the PPR restart probability to $\alpha = 0.2$, the association propagation depth to $H_A = 2$, and the maximum temporal segment length to $L_T = 3$. We use Top-10 memories as seed memories, limit the candidate pool to 60, and set the final retrieval budget to Top-10. To analyze the sensitivity of CoRelMem to different hyperparameters, we evaluate multiple parameter settings. The structural relevance weight for seed memory selection is varied over $\lambda \in \{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0\}$, and the fusion weight in IAR is varied over $\beta \in \{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0\}$. The final retrieval size is evaluated with $K_R \in \{3, 5, 10, 20, 40, 60\}$. For association retrieval, we vary the association threshold over $\tau_A \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$, the PPR restart probability over $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, and the association propagation depth over $H_A \in \{1, 2, 3\}$. For temporal retrieval, we evaluate the temporal relevance fusion weight with $\eta \in \{0, 0.25, 0.5, 0.75, 1.0\}$ and the maximum temporal segment length with $L_T \in \{1, 2, 3, 4, 5\}$.

For fair comparison, all methods are evaluated with the same backbone whenever applicable, and we align the embedding model and answer-generation settings as much as possible. LightMem, A-Mem, Mem0, and CoRelMem all use all-MiniLM-L6-v2 as the embedding model. MemGAS retains its original Contriever retriever and corresponding retrieval settings to preserve its original design. All experiments are conducted on a single NVIDIA A800 80 GB GPU. The main software environment includes Python 3.10–3.11, PyTorch 2.8.0, vLLM 0.11.0, and LLMLingua 0.2.2. Due to dependency differences, MemGAS is run in a separate Python 3.9 environment.

A.5 Additional Hyperparameter Sensitivity Analysis

A.5.1 Association Retrieval Parameters

(1) Association threshold τ_A Figure 5 examines the threshold used to retain coherent association edges. As τ_A increases, the memory association graph becomes substantially sparser: the number of edges decreases while the fraction of isolated memories increases. Nevertheless, a denser graph does not consistently yield better downstream performance, indicating that retaining excessive weak associations may introduce noisy dependencies. Among the tested settings, $\tau_A = 0.6$ provides a favorable balance between graph sparsity and QA quality, achieving the strongest F1 while maintaining competitive accuracy. We therefore use $\tau_A = 0.6$ throughout the experiments.

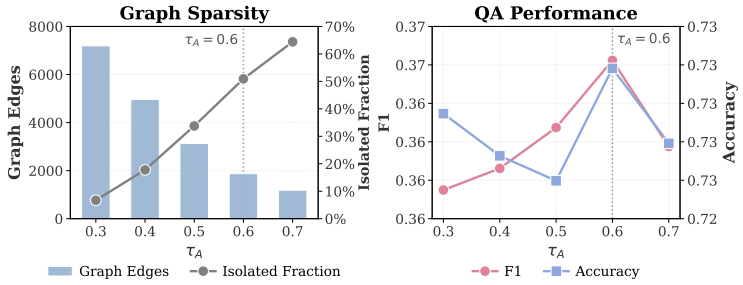


Figure 5: Sensitivity analysis of the association threshold τ_A. The left panel shows how τ_A affects the sparsity of the memory association graph, while the right reports the QA performance.

(2) PPR restart probability α Figure 6 studies the restart probability α used in Personalized PageRank over the memory association graph. Both retrieval and QA performance remain highly stable when α varies from 0.1 to 0.5. Recall, MRR, and NDCG exhibit only minor fluctuations, and the variation in F1 and accuracy is similarly small. These results show that association-based retrieval is not sensitive to the exact restart probability within a reasonable range. We therefore adopt $\alpha = 0.2$ as the default setting.

(3) Association propagation depth H_A Figure 7 investigates how far structural retrieval should propagate over the memory association graph. Increasing H_A continuously enlarges the local search space, as reflected by the growing numbers of explored nodes and edges. Expanding the neighborhood from one to two hops leads to a clear improvement in both F1 and accuracy, indicating that

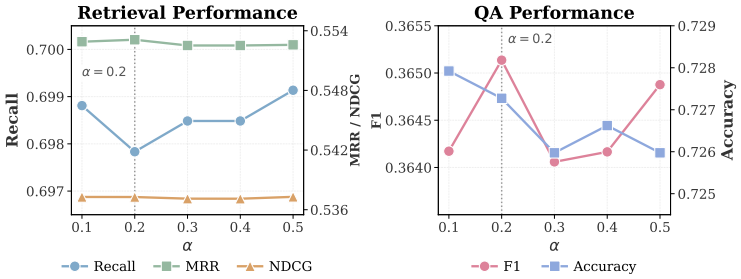


Figure 6: Sensitivity analysis of the PPR restart probability α in association-based retrieval.

direct neighbors alone may be insufficient to recover cross-memory dependencies. Further increasing the depth to three hops substantially enlarges the search scope but provides no additional QA improvement. This suggests that deeper propagation can introduce less relevant associations without providing additional useful evidence. We therefore set $H_A = 2$.

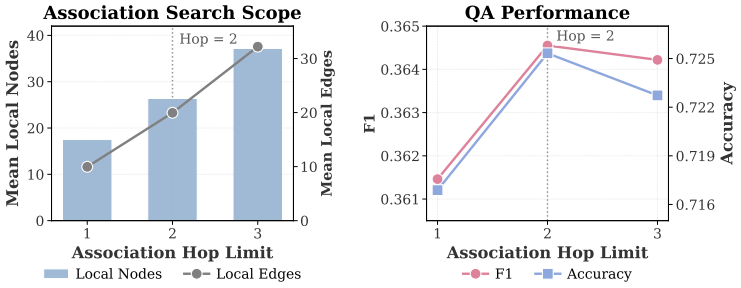


Figure 7: Sensitivity analysis of the association propagation depth H_A. The left panel reports the resulting local search scope, and the right panel shows the corresponding QA performance.

A.5.2 Temporal Retrieval Parameters

(1) Temporal relevance weight η Figure 8 analyzes the weight η that balances the maximum event relevance and the average relevance of a temporal segment. Changing η affects the selected temporal candidates for a subset of queries, especially under the extreme setting that relies entirely on the maximum event relevance. However, only a small fraction of these candidate-level changes propagate to the final Top-K_R retrieval results. Consequently, both F1 and accuracy remain nearly unchanged across the full range of η. This demonstrates that temporal retrieval is robust to the exact balance between local event salience and segment-level consistency. We set $\eta = 0.5$ to assign equal importance to the two signals.

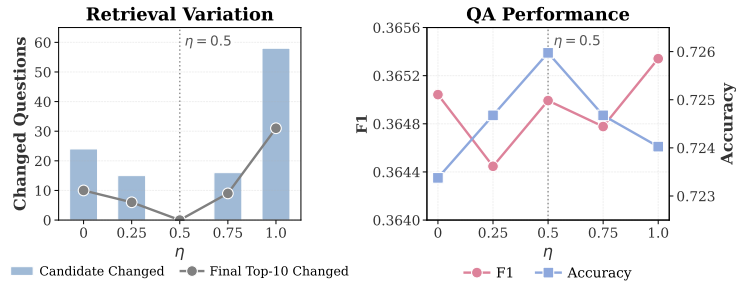


Figure 8: Sensitivity analysis of the temporal relevance weight η. The left panel shows the variation in temporal retrieval results relative to the default setting, while the right panel reports QA performance.

(2) Maximum temporal segment length L_T Figure 9 evaluates how much temporal context should be retained when constructing trajectory segments during retrieval. Increasing L_T from 1 to 2 substantially increases the effective retrieved segment length and improves QA performance, indicating that isolated temporal events may be insufficient to characterize state evolution. When L_T reaches 3, the mean selected segment length becomes nearly saturated, and further increasing the upper bound to 4 or 5 no longer changes the final Top-K_R retrieval results. The downstream QA performance also remains stable in this range. These results indicate that short contiguous segments are sufficient to capture the relevant temporal evolution while avoiding redundant events. Accordingly, we set $L_T = 3$.

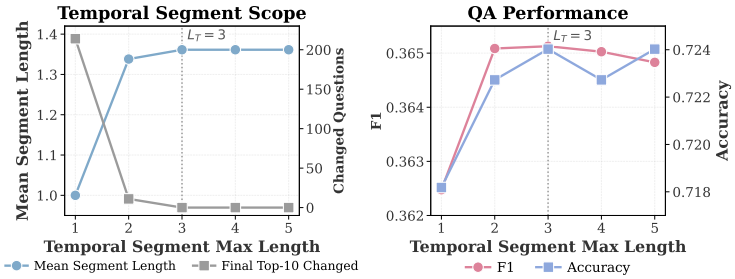


Figure 9: Sensitivity analysis of the maximum temporal segment length L_T. The left panel reports the effective temporal segment length and the variation in the final Top-K_R retrieval results, while the right panel shows QA performance.

A.6 Limitations

Although CoRelMem improves long-term memory organization and retrieval by explicitly modeling cross-memory associations and temporal evolution, it still has several limitations.

CoRelMem requires additional operations during memory construction, including atomic fact extraction, relation identification, association graph construction, and temporal trajectory organization. Compared with methods that rely only on independent memory storage and similarity-based retrieval, these operations introduce additional computational and storage overhead, which may further increase as long-term interaction histories continue to grow. Future work could explore more lightweight structure construction and incremental update mechanisms to reduce the cost of long-term memory maintenance and improve overall efficiency. Smaller models or specialized relation classifiers could also be investigated to replace part of the general-purpose LLM calls, thereby reducing the inference cost of structure construction and improving the efficiency of online memory updates.

The quality of CoRelMem's structured memory also depends on the accuracy of upstream extraction results. Errors in fact extraction, memory classification, triple generation, or relation identification may propagate to the memory association graph and temporal trajectories, thereby affecting subsequent retrieval performance. Although structured organization can enhance cross-memory reasoning, incorrect relations may also introduce additional noise. Future work could investigate more robust memory extraction and relation modeling methods, together with dynamic correction and update mechanisms, to improve the accuracy and stability of structured memory.