CORMEM: A BRAIN-INSPIRED BIPARTITE COGNITIVE MAP INDUCING LATENT RELATIONS FOR ACCURATE LONG-TERM MEMORY RETRIEVAL IN LLM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term LLM agents require memory systems that can retrieve relations across past experiences while incorporating new information without corrupting stored evidence. Relational organization plays a central role in long-term memory, where experiences are organized through associative and hierarchical structures. Current graph-based memory systems attempt to capture such relational structures. For example, association graphs capture broad semantic similarities but entangle meanings, whereas explicit-relation graphs depend on explicit relation extraction but may omit implicit relations. However, designing the representation for relationships remains challenging, as existing methods struggle to achieve both semantic explicitness and broad relational coverage. Inspired by infants' ability to infer latent relationships from statistical co-occurrence, we ask whether relational structure can be naturally induced. To this end, we propose CORMEM, a dual-layer memory that induces latent relations in a cue–fact bipartite cognitive map and forms higher-level semantic gists in a hierarchical schema layer. Cue co-occurrence creates shared-cue paths between facts, and query-conditioned Personalized PageRank (PPR) traverses this structure to activate relevant memories. Schema formation incorporates updates without rewriting stored facts. Experiments on LoCoMo and PersonaMem-v2 show consistent gains over strong baselines. Across all LoCoMo backbones, CORMEM achieves the best average F1 and improves gold-session Recall@3 by up to 18.4 points. Retrieved session co-activation also aligns with annotated co-evidence. On PersonaMem-v2, it reaches 70.5% accuracy and significantly outperforms the variant without the schema layer.

1 INTRODUCTION

Long-term memory allows LLM agents to maintain continuity, adapt to users, and reuse earlier evidence across extended interactions (Park et al., 2023; Mei et al., 2025). A useful memory must preserve what happened, expose relations across experiences, retrieve relevant evidence, and incorporate new information without corrupting earlier records.

Existing long-term memory systems adopt diverse representations, ranging from text-centric stores to graph structures. Text-centric memories retrieve stored units by similarity, attributes, recency, or importance, but relations across entities, episodes, and time points remain implicit (Park et al., 2023; Maharana et al., 2024). Graph memories make such connections traversable, yet their edge design creates a trade-off. Association graphs retain broad links from similarity, overlap, or event grouping, but their edges mix heterogeneous meanings (Xu et al., 2025; Zhang et al., 2025). Explicit-relation graphs provide typed paths, yet depend on pairwise extraction that can omit implicit relations or collapse repeated support (Gutiérrez et al., 2025; Choi et al., 2026). Memory update raises a second structural problem. Existing systems may modify the units later used as evidence through summarization, merging, rewriting, or invalidation (Chhikara et al., 2025; Rasmussen et al., 2025). Successive modifications can obscure provenance and discard atomic details.

While agent memory systems face these structural problems, biological learning offers useful functional analogies. Infant statistical learning shows that infants can learn repeated co-occurrence patterns in speech and visual scenes (Saffran et al., 1996; Fiser & Aslin, 2002). Such unlabeled

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

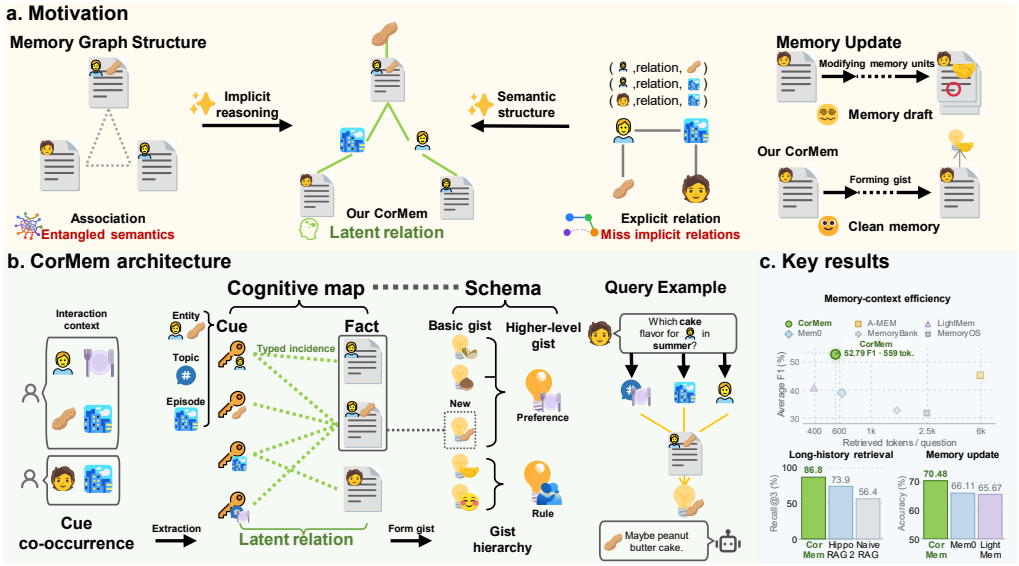

Figure 1: Overview of CorMem. **a. Motivation.** Graph memories trade relational coverage for clear edge semantics, while memory updates may rewrite evidence-bearing units. CorMem combines a cue–fact cognitive map over append-only facts with a separate schema layer. **b. Architecture.** Memory encoding, cognitive map construction, schema formation, and memory retrieval. **c. Results.** The scatter shows LoCoMo GPT-4o-mini Top-10 F1 against retrieved tokens per query. The bars show LoCoMo GPT-4o-mini Top-3 Recall@3 and PersonaMem-v2 Qwen3.5-27B query-micro accuracy.

regularities can connect otherwise separate experiences through local co-occurrences. Cognitive map theory further suggests that the resulting connections can support flexible navigation beyond fixed stimulus–response chains (Tolman, 1948). Complementary learning systems distinguish rapid encoding of episode-specific details from slower integration across experiences (McClelland et al., 1995; O'Reilly et al., 2014). Together, these findings provide a functional motivation for preserving repeated co-occurrence among intact experiences while forming slower abstractions over them.

We therefore propose CorMem (Figure 1), a dual-layer memory with append-only facts organized in a cognitive map and an evolving schema layer. The cognitive map is a cue–fact bipartite graph whose entity, topic, and episode edges state only that a fact instantiates a cue. Retaining each cue–fact incidence preserves repeated cue co-occurrence across facts as distinct, semantically grounded paths. Together, these paths induce a traversable structure of cue-mediated latent relations. Query-conditioned Personalized PageRank (PPR) traverses this structure to rank facts, while degree normalization limits the influence of ubiquitous cues. Offline schema formation organizes supported gists without rewriting facts. At answer time, a second PPR over the joint memory–schema graph ranks relevant gists.

Experiments on LoCoMo and PersonaMem-v2 support the proposed memory structure. First, the cue–fact map improves evidence retrieval. Under controlled Top-3 session retrieval, CorMem exceeds HippoRAG 2 by 12.90–18.36 Recall@3 points across three backbones. Removing propagation reduces Recall@3 by 30.42 points, while cue–fact restructuring raises HippoRAG 2's Recall@3 from 73.93% to 77.39%. Second, the induced relations are task-aligned. CorMem's session co-activation shows the highest mean alignment with annotated co-evidence ($\rho = 0.248$), and removing cues reduces alignment from 0.215 to 0.106. Third, the schema layer supports updates without rewriting facts. CorMem reaches 70.48% accuracy on PersonaMem-v2, 4.37 points above Mem0, with a 2.69-point drop without schema formation. Our contributions are threefold.

- We introduce a graph memory that organizes append-only facts in a bipartite cue–fact cognitive map and supported gists in an evolving schema layer, structurally separating fact storage from schema formation.
- We introduce cue-mediated latent relations. Typed cue–fact incidences turn repeated cue co-occurrence into shared-cue paths, which query-conditioned PPR traverses without predicting pairwise relations.

- We show that CORMEM outperforms strong baselines in long-history retrieval and update-sensitive personalization, and trace its gains to the cue–fact structure and schema layer through controlled ablations, relational analysis, and graph transfer.

2 RELATED WORK

Agent memory structure. Flat memories retain messages, notes, facts, summaries, or profiles and rank them by recency, importance, relevance, or textual and structured signals (Park et al., 2023; Zhong et al., 2024; Xu et al., 2025; Chhikara et al., 2025; Salama et al., 2025; Zhang et al., 2026a), leaving relations among records implicit (Maharana et al., 2024; Wu et al., 2025a; Tan et al., 2025). Graph memories make these relations traversable through entity–relation, temporal, or event structures (Gutiérrez et al., 2024; 2025; Rasmussen et al., 2025; Chhikara et al., 2025; Choi et al., 2026), couple episodic and semantic organization (Anokhin et al., 2025; Shu et al., 2026), associate memories through similarity, event arguments, concepts, or cue tags (Xu et al., 2025; Salama et al., 2025; Wu et al., 2025b; 2026; Zhang et al., 2025; Zhou & Han, 2025; Paul et al., 2026; Ji et al., 2026), or organize them hierarchically (Zhang et al., 2026b). MemORAI and SPRIG apply query-adaptive or seeded propagation over provenance-rich and entity–document graphs (Van et al., 2026; Wang, 2025). CORMEM differs in edge semantics. Its cognitive map has no fact–fact edges, typed cue incidences induce cue-mediated latent relations over intact source-grounded facts, and a separate schema layer holds abstractions.

Agent memory update. Memory systems add, update, rewrite, decay, delete, or invalidate stored units (Chhikara et al., 2025; Xu et al., 2025; Zhong et al., 2024; Rasmussen et al., 2025), or reorganize experience into topic, profile, or hierarchical representations (Zhang et al., 2026a; Tan et al., 2025; Ma et al., 2025; Zhang et al., 2026b). These operations improve compactness or currency but may alter evidence-bearing units. Constructivist and dual-memory agents explore related abstractions (Li et al., 2025; Shinde, 2026). CORMEM instead keeps facts append-only and confines gist formation to a separate schema layer.

3 METHOD

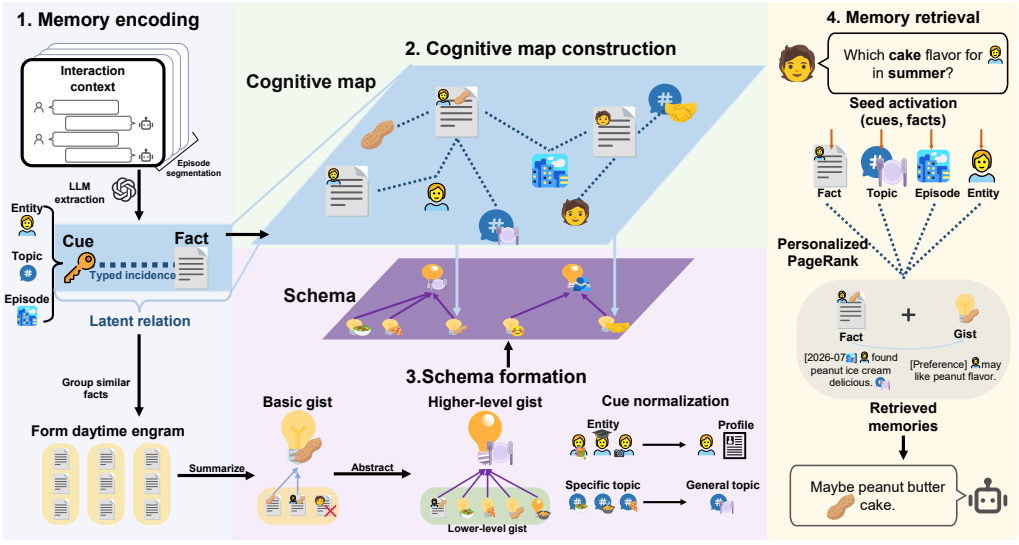

Figure 2: CORMEM workflow. Memory encoding extracts facts and cues. Cognitive map construction records typed cue–fact incidences. A cue recurring across facts creates distinct incidence paths that induce cue-mediated latent relations. Schema formation builds evidence-grounded gists without rewriting facts. Query-conditioned PPR retrieves facts and gists.

3.1 OVERVIEW

CORMEM connects past evidence through shared typed cues, making implicit relations traversable, and incorporates user updates without rewriting stored facts (Figure 2). It maintains a *Bipartite Cognitive Map* of facts $\mathcal{F}$, typed cues $\mathcal{C}$, and incidence matrix $\mathbf{B}$, plus a *gist forest* with gist nodes $\mathcal{G}$.

Memory construction encodes dialogue into the cognitive map and forms evidence-grounded gists.

$$\text{dialogue} \xrightarrow{\text{encoding}} \underbrace{(\mathcal{F}, \mathcal{C}, \mathbf{B})}_{\text{cognitive map}} \xrightarrow{\text{schema formation}} \underbrace{\mathcal{G}}_{\text{gists in a forest}} . \tag{1}$$

New facts are appended to the map. Schema formation distills fact groups into hierarchical gists. Preferences, corrections, and forgetting requests enter the schema layer, leaving facts intact.

Given retrieval context x, two Personalized PageRank (PPR) passes rank facts and gists.

$$\begin{aligned} \mathbf{r}_F(x) &= \mathrm{PPR}(\text{cognitive map}, x)|_{\mathcal{F}} , \\ \mathbf{r}_G(x) &= \mathrm{PPR}(\text{joint memory–schema graph}, x)|_{\mathcal{G}} . \end{aligned} \tag{2}$$

Here PPR returns stationary scores seeded by context-matched nodes, and $\cdot|_{\mathcal{F}}$ and $\cdot|_{\mathcal{G}}$ select fact and gist scores. Independently selected facts and gists supply evidence and a compact constraint block for answer generation. In online use, retrieval precedes insertion of the current session. Appendices B–E and K give implementation details and the complete operational prompts.

3.2 MEMORY ENCODING

Memory encoding maps one dialogue session to source-grounded facts and cues (Figure 3). We treat each session as one *episode* (mapping in Appendix D). The LLM first extracts a session-wide entity list, then extracts facts per entity, each with one topic, conditioned on that list and the full session. This entity-first order keeps facts atomic and attribution-specific. We store each fact with event and encoding times, its source-session identifier, and a source pointer, and never overwrite its text.

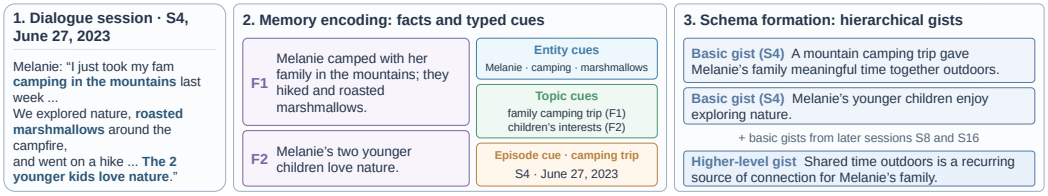

Figure 3: LoCoMo example of memory encoding and schema formation.

Within each session, semantically similar new facts form disjoint *daytime engrams*. Member facts receive *daytime memory marks*, which record an extractive trace of recurring content without replacing the facts. Singleton facts remain orphans. Each engram preserves its cue–fact subgraph and supplies the evidence from which schema formation may distill a *basic gist*.

Entity cues denote recurring participants, objects, places, activities, and concepts. *Topic cues* label a fact's thematic frame, with initial names set in the session context and later normalized across sessions. An *episode cue* connects all facts extracted from a session and retains its timestamp, summary, and dialogue context as a spatiotemporal signature. Because these cues are assigned with their facts or source session, cognitive map construction records observed typed incidences rather than predicting relations.

3.3 COGNITIVE MAP CONSTRUCTION

The *Bipartite Cognitive Map* has N fact nodes $\mathcal{F}$, which preserve source-grounded evidence, and M typed cue nodes $\mathcal{C}$, which provide entity, topic, and episode access points. Its only edges are binary cue–fact incidences, created when memory encoding assigns a cue $c_\ell \in \mathcal{C}$ to a fact $f_i \in \mathcal{F}$. There are no fact–fact or cue–cue edges. The sparse biadjacency matrix $\mathbf{B} \in \{0,1\}^{N \times M}$ is

$$B_{i\ell} = \mathbb{I}[c_\ell \in \mathrm{cue}(f_i)] . \tag{3}$$

When a cue recurs across facts, its separate incidences create fact–cue–fact paths, making relations between facts traversable without a stored fact–fact predicate. We call such a connection a *cue-mediated latent relation*. It is latent only because no direct fact–fact edge or relation label is stored, and the intervening entity, topic, or episode cue keeps its semantics inspectable.

With fact- and cue-degree matrices $\mathbf{D}_F$ and $\mathbf{D}_C$, the map induces the normalized two-step operator

$$\mathbf{T}_F = \mathbf{D}_F^{-1}\mathbf{B}\mathbf{D}_C^{-1}\mathbf{B}^\top. \tag{4}$$

$\mathbf{T}_F$ is not an additional fact–fact graph. It summarizes normalized fact–cue–fact propagation over all shared cues, down-weighting high-degree cue hubs, and memory retrieval conditions it on the query.

3.4 Schema Formation

Schema is a hierarchical forest of gist nodes $\mathcal{G}$. Each daytime engram yields an evidence-grounded *basic gist*, and each singleton fact becomes an orphan basic gist. All basic gists retain links to supporting facts. Complete-linkage clustering proposes compatible groups among current root gists, and an LLM audits whether each group supports a *higher-level gist*. Each accepted group receives one parent, preserving a single-parent forest. Repetition organizes related concepts into broader rules and preferences. Forgetting requests become rule-like constraint gists that, when retrieved, suppress revoked information while preserving source facts. This stage also forms entity profiles from entity-linked fact subgraphs and normalizes topic names across sessions to improve cue matching. The hierarchy lets higher-level gists aggregate PPR activation from multiple descendants, so a relevant abstraction can outrank any single supporting gist. Appendix D.5 details schema formation and its retrieval role.

3.5 Memory Retrieval

The retrieval context x is the query in LoCoMo and the current dialogue context at answer time in PersonaMem-v2. Hybrid BM25–embedding candidate generation and a reranker select typed cue and fact seeds matching x and assign normalized restart weights (candidate generation and reranking in Appendix E). The schema pass also restarts from a few directly matched gist seeds.

The first stage propagates cue and fact seeds only on the Bipartite Cognitive Map and reads fact scores. We deliberately use standard PPR to isolate gains from the memory structure. With normalized restart weights $\mathbf{p}_F(x)$ and $\mathbf{p}_C(x)$ over facts and cues, the stationary scores $\mathbf{r}_F(x)$ and $\mathbf{r}_C(x)$ satisfy

$$\begin{aligned}\mathbf{r}_F(x) &= \alpha\mathbf{p}_F(x) + (1-\alpha)\mathbf{B}\mathbf{D}_C^{-1}\mathbf{r}_C(x),\\ \mathbf{r}_C(x) &= \alpha\mathbf{p}_C(x) + (1-\alpha)\mathbf{B}^\top\mathbf{D}_F^{-1}\mathbf{r}_F(x).\end{aligned} \tag{5}$$

PPR repeatedly restarts from context-matched seeds and diffuses activation along cue–fact incidences, accumulating it across distinct shared-cue paths (converging routes in Figure 8, Appendix I.4). A fact supported through several context-relevant paths therefore receives more probability mass. Repeated cue occurrence across facts thus induces latent relations structurally, and PPR makes them query-conditioned at retrieval time. The highest-scoring facts form the episodic evidence output.

Schema formation supplies the cross-structure edges for gist retrieval. Each basic gist is linked bidirectionally to its supporting facts, and each child gist to its parent. The resulting *joint memory–schema graph* augments the map with gist nodes and provenance-preserving edges. The second stage runs PPR on this graph and reads only gist scores $\mathbf{r}_G(x)$ (Equation 2). Fact and gist Top-K lists are selected independently, so the gist constraint block does not displace episodic evidence.

4 Experiments

4.1 Experimental Setup

Datasets. LoCoMo (Maharana et al., 2024) measures long-history retrieval with 1,540 answerable Category 1–4 questions over ten conversations (counts and exclusions in Appendix H). PersonaMem-v2 (Jiang et al., 2025) measures causal online personalization after memory updates. Before evaluation, we selected the 100 of its 200 personas with the most queries (2,974 of 5,000 update-sensitive multiple-choice queries) to maximize coverage under a fixed construction budget.

Table 1: LoCoMo results under two retrieval protocols. **a.** Three raw sessions per query with a shared answer prompt. **b.** Ten native memory units per query. FullText reads the full history. F1 and BLEU-1 (%) average three answer generations, and R@3 is retrieval-only. Bold, underline, and italics mark the top three per backbone and panel.

a. Top-3 Session Retrieval (Common Unit)

Backbone	Method	Multi-hop		Temporal		Open-domain		Single-hop		Avg.		R@3
		F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	
GPT-4o-mini	FullText	33.42	21.88	29.15	24.15	*16.27*	*12.27*	*44.79*	36.85	37.67	29.93	–
	NaiveRAG	*33.51*	*22.64*	*36.74*	*31.91*	15.02	10.50	44.56	*39.14*	*39.07*	*32.83*	*56.41*
	HippoRAG 2	38.96	27.30	48.11	42.87	**20.84**	**15.77**	48.77	41.13	45.09	37.38	73.93
	CorMem	**41.51**	**29.44**	**57.10**	**51.25**	19.51	15.14	**64.31**	**56.70**	**55.84**	**47.98**	**86.83**
Qwen3.5-27B	FullText	46.11	33.47	57.82	52.00	22.71	18.94	59.85	52.59	54.60	46.87	–
	NaiveRAG	38.36	27.57	40.44	35.69	*25.01*	*20.94*	49.12	43.98	43.84	37.81	*56.66*
	HippoRAG 2	*38.57*	*31.04*	*48.47*	*43.19*	25.54	22.11	*53.48*	*47.34*	*47.96*	*41.91*	69.40
	CorMem	**49.08**	**36.73**	**60.08**	**53.25**	**30.98**	**26.37**	**70.99**	**63.24**	**62.21**	**54.01**	**87.76**
DeepSeek-v4-Flash	FullText	*35.57*	*28.36*	45.33	*36.59*	23.10	19.09	58.21	50.07	49.19	41.35	–
	NaiveRAG	37.18	27.13	*41.68*	36.96	27.86	23.70	47.84	42.39	43.36	37.30	*56.66*
	HippoRAG 2	34.93	28.65	32.63	26.80	*23.74*	*19.44*	*53.90*	*47.48*	*44.11*	*37.97*	71.46
	CorMem	**48.20**	**35.09**	**58.76**	**52.76**	**29.30**	**24.52**	**70.00**	**62.54**	**61.13**	**53.11**	**88.12**

b. Top-10 Memory-Unit Retrieval (Method-Specific Units)

Backbone	Method	Multi-hop		Temporal		Open-domain		Single-hop		Avg.	
		F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1
GPT-4o-mini	FullText	33.42	21.88	29.15	24.15	16.27	12.27	44.79	36.85	37.67	29.93
	NaiveRAG	23.15	14.41	8.92	6.02	14.39	11.07	35.86	30.83	26.58	21.42
	MemoryBank	34.70	22.63	*32.17*	*27.47*	**19.75**	**15.17**	33.93	28.18	32.82	26.20
	A-MEM	*38.75*	25.36	50.74	45.34	16.34	12.02	47.05	39.68	44.38	36.51
	Mem0	37.13	*26.07*	19.01	16.41	18.83	13.96	*48.92*	*41.67*	38.65	31.82
	MemoryOS	30.06	20.45	17.86	14.96	15.64	11.60	38.61	32.36	31.29	25.26
	LightMem	**45.40**	30.62	20.58	15.08	*18.61*	*13.76*	49.65	42.14	*40.88*	*32.62*
	CorMem	44.27	**30.74**	**56.90**	**51.12**	18.33	13.01	**58.00**	**49.87**	**52.79**	**44.33**
Qwen3.5-27B	FullText	46.11	**33.47**	**57.82**	**52.00**	22.71	18.94	59.85	52.59	54.60	46.87
	NaiveRAG	17.60	11.94	5.56	3.09	11.39	8.92	35.75	30.86	24.61	20.24
	MemoryBank	33.62	24.17	31.62	27.34	17.14	14.11	33.45	28.41	32.08	26.52
	A-MEM	*45.00*	*32.64*	*56.12*	*49.90*	*21.20*	*18.05*	*54.43*	*47.66*	*50.98*	*43.53*
	Mem0	40.41	29.70	8.52	7.20	17.96	14.97	52.37	45.14	38.89	32.52
	MemoryOS	32.41	23.54	17.63	14.53	17.33	14.29	38.21	33.23	31.55	26.38
	LightMem	36.33	27.70	24.62	20.74	18.03	14.76	49.04	41.67	39.69	33.07
	CorMem	**46.66**	33.29	57.00	50.80	**30.23**	**25.94**	**63.58**	**55.23**	**57.03**	**48.46**
DeepSeek-v4-Flash	FullText	*35.57*	*28.36*	*45.33*	*36.59*	*23.10*	19.09	58.21	50.07	49.19	*41.35*
	NaiveRAG	21.76	15.41	10.82	7.12	20.82	16.54	45.14	38.31	32.19	26.26
	MemoryBank	29.60	21.30	29.11	24.18	18.51	14.41	36.74	30.79	32.71	26.65
	A-MEM	40.39	**30.79**	39.41	32.80	21.59	17.16	60.12	53.08	49.79	42.53
	Mem0	34.44	27.05	14.09	12.36	23.16	18.02	*59.17*	*51.28*	43.00	36.66
	MemoryOS	29.12	22.27	14.42	10.37	22.40	*18.61*	49.83	42.97	36.95	30.87
	LightMem	34.89	26.93	**58.43**	**51.96**	20.72	15.89	53.92	44.74	*49.30*	41.18
	CorMem	**43.79**	30.77	48.47	43.10	**27.97**	**23.09**	**64.08**	**55.69**	**54.86**	**46.47**

Baselines. HippoRAG 2 (Gutiérrez et al., 2025) is the closest structural comparison. Both build maps from sessions, use PPR, and return raw sessions in the controlled protocol, leaving graph representation as the difference. We include FullText, NaiveRAG, MemoryBank (Zhong et al., 2024), A-MEM (Xu et al., 2025), Mem0 (Chhikara et al., 2025), MemoryOS (Kang et al., 2025), and LightMem (Zhang et al., 2026a), using official or author-released code with a shared answer prompt.

Protocol. On LoCoMo, the controlled Top-3 protocol gives NaiveRAG, HippoRAG 2, and CorMem three raw sessions under the same answer prompt, with CorMem mapping fact scores to source sessions. Fixing the returned unit, budget, and prompt isolates evidence ranking. A complementary Top-10 method-native protocol tests whether the gains persist with each method's native memory units. In both settings, CorMem uses only cognitive-map facts, without schema

gists, entity profiles, or source-dialogue expansion. PersonaMem-v2 is evaluated online using only sessions preceding each query, and CORMEM retrieves up to ten facts and two gists.

Metrics. On LoCoMo, F1 is the token-overlap F1 between the generated and reference answers after normalization and stemming, and BLEU-1 is the unigram BLEU score, both averaged over three answer generations. Retrieval-only gold-session Recall@3 is $|G(q) \cap R_3(q)|/|G(q)|$, where $G(q)$ is the set of annotated gold sessions and $R_3(q)$ is the set of three returned sessions, over the 1,536 questions with gold-session evidence. PersonaMem-v2 reports query-micro accuracy, the fraction of queries whose selected option matches the gold option. Paired tests and bootstrap confidence intervals are clustered by conversation or persona (Appendix H).

Implementation details. The LoCoMo answer backbones are GPT-4o-mini, Qwen3.5-27B, and DeepSeek-v4-Flash, and PersonaMem-v2 uses Qwen3.5-27B. Each backbone also performs fact/cue extraction, gist summarization, and auditing. Appendices B–H detail the full setup and token-matched comparisons, and Appendix I.6 reports hyperparameter sensitivity.

4.2 MAIN RESULTS

LoCoMo: Controlled Session Retrieval. CORMEM recovers evidence-bearing sessions more reliably than the closest explicit-relation graph baseline. In Table 1a, it exceeds HippoRAG 2 by 12.90–18.36 Recall@3 points and 10.75–17.02 average F1 points across three backbones. F1 improves across the stored-evidence categories except GPT-4o-mini open-domain questions, which bounds the claim to dialogue evidence. Without session-derived episode cues, Recall@3 drops by 0.76 points but remains 12.17 points above HippoRAG 2 ($p = .002$). Entity, topic, and episode seeds all contribute to the PPR scores of retrieved gold sessions (Appendix I.3, Figure 7).

LoCoMo: Native-Unit Retrieval. In Table 1b, CORMEM has the highest native-unit average F1 on every backbone, 5.07–8.41 points above the strongest memory baseline. Different unit content and compression preclude pure structural attribution, which is reserved for the controlled panel.

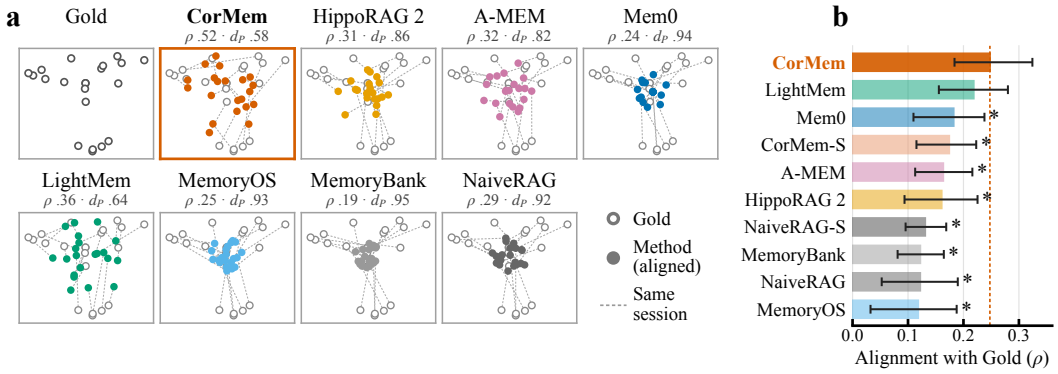

Figure 4: Relational organization on LoCoMo (GPT-4o-mini). **a.** MDS maps of conv-47 (21 sessions), the conversation with the largest CORMEM margin. Gray circles are gold positions, filled points are Procrustes-aligned method positions, and dashed lines join the same session. d_P: Procrustes disparity. **b.** Mean ρ over ten conversations with 95% cluster bootstrap CIs. Suffix -S marks the Top-3 session readout. $^*p < .05$ versus CORMEM (exact paired sign-flip, unadjusted).

LoCoMo: Relational Organization. We next ask whether the relational structure induced by the cognitive map is organized around the task. If cue-mediated latent relations connect related experiences, normal retrieval should co-activate the sessions that jointly support a question, even though no session pairing is stored. We test this on the frozen GPT-4o-mini retrievals of Table 1 without new retrieval. For each conversation, the annotated co-evidence matrix $G_{ij} = C_{ij}/\sqrt{f_i f_j}$ normalizes how often sessions i and j are gold evidence for the same question. For each method, a question–session indicator marks the sessions it returns under its main-table protocol, namely Top-3 sessions (Table 1a) or the source sessions of Top-10 native units (Table 1b). The association A_{ij} is the correlation of sessions

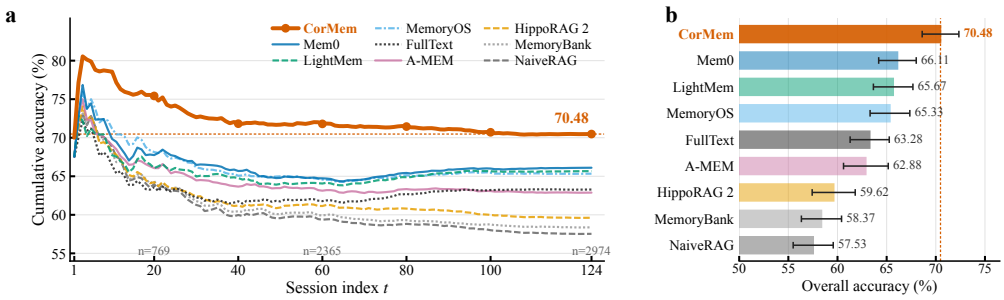

Figure 5: Online personalization on PersonaMem-v2 (Qwen3.5-27B, 2,974 queries). **a.** Cumulative accuracy over queries with session index at most t. **b.** Overall accuracy with 95% persona-cluster bootstrap CIs. The right endpoints of (a) equal (b).

i and j across the 1,536 questions with gold-session evidence. Following representational similarity analysis (Kriegeskorte et al., 2008), alignment is the Spearman correlation between the upper triangles of G and A, averaged over conversations. For visualization, rank-based session distances are embedded by classical MDS (Torgerson, 1952) and Procrustes-aligned to the gold map (Gower, 1975).

CorMem shows the highest alignment with annotated co-evidence (Figure 4b). Its mean alignment is $\rho = 0.248$, above LightMem (0.218) and HippoRAG 2 (0.161). It exceeds eight of nine comparisons in unadjusted paired conversation-cluster tests ($p \leq .043$), while the LightMem difference is not significant ($p = .266$). In conv-47 (Figure 4a), CorMem preserves the spread of the gold session clusters ($\rho = 0.52$, $d_P = 0.58$), whereas most baselines collapse toward the center. Because A is measured under task queries, this analysis characterizes query-conditioned co-activation rather than a query-independent session graph. Definitions and per-conversation values are given in Appendix I.2 of the supplementary material. Under a matched Top-3 session budget, CorMem remains among the strongest methods ($\rho = .223$), statistically tied with LightMem (.222) and above the remaining baselines. No cue drops alignment (0.215 to 0.106 on Top-10 facts, 0.174 to 0.090 on Top-3 sessions, $p < .035$). w/o Episode has little effect (0.215/0.181). See Appendix I.2.

PersonaMem-v2: Personalized updates. CorMem improves causal online personalization over the strongest baseline. It reaches 70.48% query-micro accuracy in Figure 5b, 4.37 points above Mem0 ($p < .001$, paired persona-cluster sign-flip). The prefix curve in Figure 5a stays above every baseline as interaction deepens. Persona-macro accuracy is 70.42% with the same ranking.

4.3 Ablation Study

Cognitive map structure. Full bipartite propagation accounts for the controlled retrieval advantage. Removing propagation in Table 2 lowers Recall@3 by 30.42 points, while fact-only and cue-only projections lose 4.04 and 8.43 points. Fact-only links facts sharing a cue and uses fact seeds, whereas cue-only links cues co-occurring in a fact and uses cue seeds (projected weights in Appendix F). With the pipeline otherwise fixed and Recall@3 independent of answer generation, these drops attribute the Table 1a advantage to graph traversal over the full bipartite topology (largest Holm-adjusted $p = .006$).

Graph-structure transfer. We test the retrieval benefit of a cue–fact bipartite backbone by restructuring HippoRAG 2's graph. Each OpenIE triple becomes a fact node linked to its subject and object phrases as cues and to its source session, replacing triple-derived edges. Facts sharing cues are thus joined by fact–cue–fact paths along which PPR aggregates evidence. We retain the models, PPR parameters, and Top-3 budget (conversion details and statistics in Appendix I.1). Recall@3 rises from 73.93% to 77.39% ($p_{\mathrm{H}} = .004$). This significant gain supports the retrieval value of organizing co-occurring evidence through bipartite cue–fact incidences. CorMem reaches 86.83% (Figure 6).

Schema formation mechanism. Schema formation improves updates more reliably than rewriting. In Table 3, it raises PersonaMem accuracy from 67.79% to 70.48% ($p < .001$), a 2.69-point gain versus 1.31 for Rewrite (69.10%). On diagnostic LoCoMo, it raises F1 to 54.31, whereas Rewrite lowers it to 51.56 with more regressions (24.29% versus 15.71%). These ablations support schema formation as safer than rewriting evidence-bearing facts (update example in Appendix I.5). Context

Table 2: Cognitive-map ablation on LoCoMo (GPT-4o-mini, Top-3 sessions). ‡Higher R@3 than every variant (paired cluster tests, Holm $p \leq .006$).

Variant	F1	BLEU-1	R@3
w/o cue (fact-only)	55.24	47.40	82.79
w/o fact (cue-only)	52.44	45.02	78.40
w/o graph (flat)	40.22	34.06	56.41
CorMem	**55.84**	**47.98**	**86.83**‡

Table 3: Update ablation. PFR and NFR are positive and negative flip rates relative to w/o schema. LoCoMo is diagnostic. †$p < .001$ versus w/o schema (paired persona-cluster test).

	LoCoMo			PersonaMem		
Variant	**F1**	**PFR↑**	**NFR↓**	**Acc.**	**PFR↑**	**NFR↓**
w/o schema	53.85	–	–	67.79	–	–
w/ schema	**54.31**	16.36	15.71	**70.48**†	7.50	4.81
Rewrite	51.56	18.77	24.29	69.10	5.21	3.90

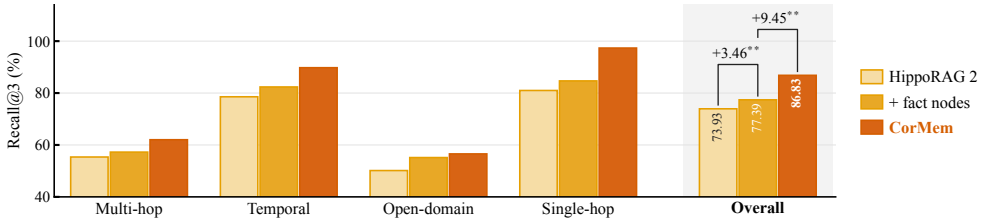

Figure 6: Bipartite restructuring of HippoRAG 2 on LoCoMo (GPT-4o-mini, n=1,536). Brackets mark Overall gains. **Holm-adjusted p=.004 (exact paired cluster tests). The axis starts at 40%.

length does not explain the gain. Ten facts plus two gists average 749.86 tokens versus 620.43 for ten facts, yet token-matched Top-12 facts reach only 67.65% ($p = .868$ versus Top-10).

Table 4: Retrieval cost, task performance, and end-to-end construction cost. Tok./Q counts retrieved memory tokens passed to the answer model per query. Build gives the mean LLM calls and input/output tokens to construct memory for one LoCoMo conversation. F1 is LoCoMo Top-10 with GPT-4o-mini, and Acc. is PersonaMem-v2 with Qwen3.5-27B.

	LoCoMo		PersonaMem-v2		Build / Conv.		
Method	**Tok./Q ↓**	**F1 ↑**	**Tok./Q ↓**	**Acc. ↑**	**Calls ↓**	**Input (k) ↓**	**Output (k) ↓**
MemoryBank	1526.9	32.82	3141.6	58.37	–	–	–
A-MEM	5948.0	44.38	2194.1	62.88	1790.2	*1126.6*	*83.4*
Mem0	*622.4*	38.65	**375.3**	66.11	602.0	5316.6	74.2
MemoryOS	2486.6	31.29	999.8	65.33	1655.2	522.0	165.4
LightMem	**252.1**	*40.88*	376.2	*65.67*	**115.0**	**120.1**	**21.0**
CorMem	559.1	**52.79**	*725.4*	**70.48**	*1190.1*	2001.5	240.5

4.4 Efficiency Analysis

CorMem uses 559.1 LoCoMo and 725.4 PersonaMem-v2 tokens per query, below MemoryBank, A-MEM, and MemoryOS (Table 4). LightMem uses fewer tokens in both settings, and Mem0 on PersonaMem-v2. Larger-context systems score lower, so context volume alone does not explain the gains. LightMem has the lowest construction cost, while CorMem has the highest F1 and accuracy.

5 Conclusion

Shared cues induce latent relations in CorMem's bipartite cue–fact cognitive map without predicting pairwise relations. Query-conditioned PPR traverses these paths to retrieve evidence. A separate schema layer incorporates updates while keeping facts append-only. Experiments support improved evidence retrieval, task-aligned relational organization, and update-sensitive personalization. Future work will expand comparisons with recent structured-memory systems (Appendix I.8) and study construction efficiency, access control, privacy-preserving storage, and verifiable forgetting.

AI USE STATEMENT

We used generative AI tools to support literature search and review, track recent research, and assist with code development. We also used these tools to polish the authors' original manuscript and improve its clarity and concision to meet the nine-page limit. The authors are responsible for the final text, code, citations, and scientific claims.

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*, pp. 12–20. International Joint Conferences on Artificial Intelligence Organization, 2025. doi: 10.24963/ijcai.2025/2. URL https://www.ijcai.org/proceedings/2025/0002.

Shuqi Cao, Jingyi He, and Fei Tan. Higmem: A hierarchical and llm-guided memory system for long-term conversational agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 33853–33862, 2026.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. In *ECAI 2025 – 28th European Conference on Artificial Intelligence*, volume 413 of *Frontiers in Artificial Intelligence and Applications*, pp. 2993–3000. IOS Press, 2025. doi: 10.3233/FAIA251160. URL https://doi.org/10.3233/FAIA251160.

Minjun Choi, Yoonjin Jang, Sangwon Youn, and Youngjoong Ko. G-Long: Graph-enhanced memory management for efficient long-term dialogue agents, 2026. URL https://arxiv.org/abs/2606.13115.

József Fiser and Richard N. Aslin. Statistical learning of new visual feature combinations by infants. *Proceedings of the National Academy of Sciences of the United States of America*, 99(24):15822–15826, 2002. doi: 10.1073/pnas.232472899. URL https://doi.org/10.1073/pnas.232472899.

J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975. doi: 10.1007/BF02291478.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569. Curran Associates, Inc., 2024. doi: 10.52202/079017-1902. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/6ddc001d07ca4f319af96a3024f6dbd1-Paper-Conference.pdf.

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

Shuo Ji, Yibo Li, and Bryan Hooi. Memory is reconstructed, not retrieved: Graph memory for LLM agents, 2026. URL https://arxiv.org/abs/2606.06036.

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, Radha Poovendran, Gregory Wornell, Lyle Ungar, Dan Roth, Sihao Chen, and Camillo Jose Taylor. PersonaMem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory, 2025. URL https://arxiv.org/abs/2512.06688.

Dongming Jiang, Yi Li, Guanpeng Li, and Bingzhe Li. Magma: A multi-graph based agentic memory architecture for ai agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 36848–36865, 2026a.

Hanqi Jiang, Junhao Chen, Yi Pan, Ling Chen, Weihang You, Yifan Zhou, Ruidong Zhang, Yohannes Abate, and Tianming Liu. Synapse: Empowering llm agents with episodic-semantic memory via spreading activation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 22020–22036, 2026b.

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, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1318. URL https://aclanthology.org/2025.emnlp-main.1318/.

Nikolaus Kriegeskorte, Marieke Mur, and Peter Bandettini. Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2:4, 2008. doi: 10.3389/neuro.06.004.2008.

Rui Li, Zeyu Zhang, Xiaohe Bo, Zihang Tian, Xu Chen, Quanyu Dai, Zhenhua Dong, and Ruiming Tang. CAM: A constructivist view of agentic memory for LLM-based reading comprehension. In *Advances in Neural Information Processing Systems*, volume 38, pp. 113381–113406. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a488aa1a0c00d76db8a922ef7815a786-Paper-Conference.pdf.

Wenquan Ma, Jiayan Nan, Wenlong Wu, and Yize Chen. What deserves memory: Adaptive memory distillation for LLM agents, 2025. URL https://arxiv.org/abs/2508.03341.

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, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.747. URL https://aclanthology.org/2024.acl-long.747/.

James L. McClelland, Bruce L. McNaughton, and Randall C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995. doi: 10.1037/0033-295X.102.3.419. URL https://doi.org/10.1037/0033-295X.102.3.419.

Kai Mei, Xi Zhu, Wujiang Xu, Mingyu Jin, Wenyue Hua, Zelong Li, Shuyuan Xu, Ruosong Ye, Yingqiang Ge, and Yongfeng Zhang. AIOS: LLM agent operating system. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=L4HHkCDz2x.

Randall C. O'Reilly, Rajan Bhattacharyya, Michael D. Howard, and Nicholas Ketz. Complementary learning systems. *Cognitive Science*, 38(6):1229–1248, 2014. doi: 10.1111/j.1551-6709.2011.01214.x. URL https://doi.org/10.1111/j.1551-6709.2011.01214.x.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–22, New York, NY, USA, 2023. Association for Computing Machinery. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

Swarna Kamal Paul, Shubhendu Sharma, and Nitin Sareen. GAAMA: Graph augmented associative memory for agents, 2026. URL https://arxiv.org/abs/2603.27910.

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

Jenny R. Saffran, Richard N. Aslin, and Elissa L. Newport. Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928, 1996. doi: 10.1126/science.274.5294.1926. URL https://doi.org/10.1126/science.274.5294.1926.

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, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1683. URL https://aclanthology.org/2025.emnlp-main.1683/.

Zhanyu Shen, Sijie Cheng, Zhicheng Guo, Weiqin Wang, Yile Wang, and Hui Huang. Anchormem: Anchored facts with associative contexts for building memory in large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34784–34798, 2026.

Saish Sachin Shinde. SCM: Sleep-consolidated memory with algorithmic forgetting for large language models, 2026. URL https://arxiv.org/abs/2604.20943.

Yiheng Shu, Saisri Padmaja Jonnalagedda, Xiang Gao, Bernal Jiménez Gutiérrez, Weijian Qi, Kamalika Das, Huan Sun, and Yu Su. REMem: Reasoning with episodic memory in language agent. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10008195.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long T. Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, Anand Rajan Iyer, Tianlong Chen, Huan Liu, Chen-Yu Lee, and Tomas Pfister. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8416–8439, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.413. URL https://aclanthology.org/2025.acl-long.413/.

Edward C. Tolman. Cognitive maps in rats and men. *Psychological Review*, 55(4):189–208, 1948. doi: 10.1037/h0061626. URL https://doi.org/10.1037/h0061626.

Warren S. Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952. doi: 10.1007/BF02288916.

Hung Pham Van, Nguyen Manh Hieu, Khang Pham Tran Tuan, Nam Le Hai, Linh Ngo Van, Nguyen Thi Ngoc Diep, and Trung Le. MemORAI: Memory organization and retrieval via adaptive graph intelligence for LLM conversational agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 28235–28253, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1408. URL https://aclanthology.org/2026.findings-acl.1408/.

Qizhi Wang. Democratizing GraphRAG: Linear, CPU-only graph retrieval for multi-hop QA, 2025. URL https://arxiv.org/abs/2602.23372.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=pZiyCaVuti.

Yaxiong Wu, Yongyue Zhang, Sheng Liang, and Yong Liu. SGMem: Sentence graph memory for long-term conversational agents, 2025b. URL https://arxiv.org/abs/2509.21212.

Zhaofen Wu, Hanrong Zhang, Fulin Lin, Wujiang Xu, Xinran Xu, Yankai Chen, Shaowen Chen, Henry Peng Zou, Weizhi Zhang, Xue Liu, Philip S. Yu, and Hongwei Wang. GAM: Hierarchical graph memory for LLM-based agents. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mmsVZGaYyp.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, pp. 17577–17604. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/19909c36f51abc4856b4560aff3d36d6-Paper-Conference.pdf.

Jiaquan Zhang, Chaoning Zhang, Shuxu Chen, Zhenzhen Huang, Pengcheng Zheng, Zhicheng Wang, Ping Guo, Fan Mo, Sung-Ho Bae, Jie Zou, Jiwei Wei, and Yang Yang. 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, San Diego, California, United States, 2026a. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.588. URL https://aclanthology.org/2026.acl-long.588/.

Ningning Zhang, Xingxing Yang, Zhizhong Tan, Weiping Deng, and Wenyong Wang. HiMem: Hierarchical long-term memory for LLM long-horizon agents, 2026b. URL https://arxiv.org/abs/2601.06377.

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, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.901. URL https://aclanthology.org/2025.findings-acl.901/.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19724–19731, 2024. doi: 10.1609/aaai.v38i17.29946. URL https://doi.org/10.1609/aaai.v38i17.29946.

Sizhe Zhou and Jiawei Han. A simple yet strong baseline for long-term conversational memory of LLM agents, 2025. URL https://arxiv.org/abs/2511.17208.

Supplementary Material

A Reproducibility Scope

Appendices B–E detail model settings, memory construction, schema formation, and retrieval. Appendices F–H specify ablation controls, baseline implementations, evaluation protocols, end-to-end construction costs, and statistical tests.

Appendix I presents graph-structure transfer, relational organization, retrieval diagnostics, update cases, and hyperparameter sensitivity, followed by reported comparisons with recent memory systems in Appendix I.8. Appendix J lists token-matched reproducibility artifacts, and Appendix K provides the complete operational prompts.

B Models and Inference Configuration

B.1 Model Assignment

Table 5 gives the model assignment for each reported setting. For LoCoMo, each store is independently constructed with the same backbone used to answer that row. Every reported setting uses the same three-stage encoder interface in Appendix C: session-wide entity discovery, per-entity fact extraction, and session-summary/episode-cue construction. The stages write the persistent fact, cue, episode, and schema fields described in Appendix D; the complete encoding templates are in Appendix K.1.

Table 5: Stage-level model assignment. "Auditing" includes intra-session consolidation and schema grouping decisions. All rows use `text-embedding-3-large` and the pretrained `BAAI/bge-reranker-v2-m3`, without task-specific fine-tuning.

Setting	**Construction and auditing configuration**	**Answer model**
LoCoMo, GPT	GPT-4o-mini; three-stage	GPT-4o-mini
LoCoMo, Qwen	Qwen3.5-27B; three-stage	Qwen3.5-27B
LoCoMo, DeepSeek	DeepSeek-v4-Flash; three-stage	DeepSeek-v4-Flash
PersonaMem-v2	Qwen3.5-27B; three-stage	Qwen3.5-27B

The controlled LoCoMo Top-3 output contains three raw source sessions and does not inject schema text. Its method-native Top-10 output contains compact facts only. The PersonaMem-v2 main setting contains up to ten facts and two causal schema gists. In all cases, "profile disabled" or "source dialogue disabled" refers to text passed to the answer model; derived cue representations remain available for candidate matching and PPR seeding.

B.2 Inference Configuration

All generative calls use the model assignments in Table 5; embedding and reranking use the model identifiers stated in its caption. Qwen3.5-27B and DeepSeek-v4-Flash run with model-side thinking disabled. Construction calls request JSON objects at temperature 0; answer-generation requests retain the frozen benchmark-specific settings. Table 6 records the request fields and output contracts used by each call.

Table 6: Request settings. "Omitted" means that the field was not sent in the frozen answer request, rather than being assigned an implicit numerical value by the experiment code.

Call	Temperature	Output cap	Output contract
Stage 1 entity discovery	0	4,096	JSON entity inventory with source turns
Stage 2 fact extraction	0	4,096	JSON facts with topic, time, entities, and evidence
Stage 3 session summary	0	1,024	One JSON session summary with time and evidence
Intra-session consolidation	0	2,048	JSON partition; every candidate index appears exactly once
Schema audits and gist synthesis	0	2,048	JSON decision or structured gist object
Entity profile construction	0.2	2,500	JSON chronological chain and memory snapshot
LoCoMo repeated answers	omitted	omitted	Prompt-defined JSON; response-format field omitted
PersonaMem-v2 multiple choice	omitted	900	JSON with one choice in {A,B,C,D} and a brief reason

C Prompt Interfaces and Output Handling

This appendix records the model-call sequence and output handling used by the reported implementation. The complete operational templates, including every rule, run-time placeholder, output schema, and format example, are reproduced together in Appendix K.

C.1 Memory-Construction Interface

The encoder processes one timestamped session in three stages: session-wide entity discovery, per-entity fact extraction, and one session summary. The per-entity outputs are unioned and consolidated within the same session. After the summary is returned, the program deterministically assembles the episode cue from that summary, the session timestamp, raw turns, and committed fact identifiers; no additional generation call is used. The complete encoding and consolidation templates are in Appendix K.1.

C.2 Schema-Construction Interface

Schema formation separates level-1 coherence auditing, basic-gist synthesis, higher-level grouping, and higher-level gist synthesis. Entity-profile construction consumes one entity's time-sorted connected facts and evidence. The complete schema and entity-profile templates are in Appendix K.2.

C.3 Answer and Rewrite Interfaces

LoCoMo and PersonaMem-v2 use benchmark-specific answer contracts while changing only the serialized method-specific memory block within each comparison. Their complete answer templates are in Appendix K.3. The Rewrite comparison uses a separate update-decision interface whose complete rules, input and output contract, and format example are in Appendix K.4.

C.4 Output Validation

Each construction output is schema-validated before commit; an empty, malformed, or incomplete object does not modify persistent state. A partition must cover every candidate exactly once, and a declined higher-level merge leaves the existing roots unchanged.

Table 7: Persistent fields of a source-grounded fact.

Field	Meaning
`fact_id`, `fact`	Content-derived identifier and source-grounded atomic text.
`fact_kind`, `entities`	Single-entity or interaction record and its entity-cue identifiers.
`semantic_topic`	The single topic cue assigned during extraction.
`time.event_time`	Normalized point, range, mixed interval, or unknown time with confidence and reason.
`time.created_at`, `time.last_modified_at`	Encoding timestamps; committed fact text remains append-only.
`source_session_id`	Source episode used for provenance and session-return evaluation.
`evidence`, `evidence_context`	Source turn identifiers and the preserved local dialogue windows.
`confidence`, `observation_history`	Extraction confidence and an audit field retained by the record format.

D Cognitive Map Construction and Schema Formation

D.1 Dataset to Episode Mapping

The benchmark's session boundary is the episode boundary. No learned subdivision is applied. For each session s, the loader preserves the session identifier, ordinal position, timestamp, speaker-tagged turns, and image descriptions supplied by the benchmark. It constructs one episode cue e_s containing the session identifier, timestamp, summary, raw turns, and the identifiers of facts whose source session is s. The episode representation used for BM25 and embedding matching concatenates the timestamp, session summary, and raw dialogue. The cognitive-map incidence for e_s connects it only to facts extracted from s; there are no stored episode–episode edges.

Both benchmarks use the three-stage entity-first sequence in Appendix C, with complete templates in Appendix K.1, and write the same persistent record types. A candidate is consolidated with same-session duplicates before it becomes a persistent fact. After commitment, later sessions append new fact records and never replace the text of a previously committed fact.

D.2 Persistent Fact Record

Each committed fact contains the fields in Table 7. A content-derived identifier supports deduplication. Event time records when the content holds; encoding time records when the source session entered memory. The source-session identifier and evidence turns determine the unit fact–session output incidences used by the controlled LoCoMo protocol.

Let $h(s)$ denote the first 16 hexadecimal characters of $\mathrm{SHA256}(s)$. Identifiers are deterministic:

$$\texttt{fact_id} = \texttt{F_} \| h(\texttt{fact}),$$
$$\texttt{entity_id} = \texttt{E_} \| h(\mathrm{lower}(\mathrm{trim}(\texttt{name}))).$$

Repeated serialization is therefore idempotent without changing the source-grounded text.

For same-session consolidation, candidate embeddings are computed from $\texttt{topic} \parallel \texttt{fact}$. Two candidates enter the same connected component when they have identical entity sets and cosine similarity at least 0.92, or when their similarity is at least 0.97 regardless of entity set. The LLM then applies the consolidation prompt in Appendix K.1. This operation is pre-commit deduplication, not cross-session rewriting.

D.3 Bipartite Cognitive Map

Let $\mathcal{F}$ contain the committed facts and let $\mathcal{C} = \mathcal{C}_E \cup \mathcal{C}_T \cup \mathcal{C}_S$ contain entity, topic, and episode cues. For every fact f_i, the graph builder obtains its entity identifiers and topic label from the fact record and its episode identifier from `source_session_id`. It stores one undirected, unweighted edge

for each observed assignment. Hence

$$B_{i\ell} = 1 \quad \Longleftrightarrow \quad c_\ell \in \mathrm{cue}(f_i),$$

and repeated serialization of the same assignment does not increase its weight. The graph stores neither fact–fact nor cue–cue edges.

If f_i and f_j share cue c_ℓ, their two incidences create the inspectable path $f_i \to c_\ell \to f_j$. If they share several cues, each cue contributes a distinct path. Degree normalization gives the two-step fact operator

$$\mathbf{T}_F = \mathbf{D}_F^{-1}\mathbf{B}\mathbf{D}_C^{-1}\mathbf{B}^\top,$$

which is computed from the bipartite graph at retrieval time and is not stored as a separate fact graph. The $\mathbf{D}_C^{-1}$ term spreads the contribution of a high-degree cue over its incident facts, limiting cue hubs without an additional learned edge weight.

D.4 Cue Maintenance

The aliasing contract permits a merge only when the available evidence indicates that two mentions refer to the same participant or object. For each entity cue, the system orders its connected facts by event time and constructs a compact profile containing a chronological narrative chain and a memory snapshot. Missing profiles are constructed before the first query that can retrieve the committed store. When a completed session adds facts for an entity, its cached profile is marked stale and reconstructed from all of its time-sorted connected facts before the next query. The episode and cognitive-map indexes are then rebuilt, so retrieval uses the refreshed profile and graph snapshot together. Exact name or alias matches are used first at retrieval; profile matching fills unused entity-seed slots. Entity rarity is recorded as

$$\mathrm{idf}(c) = \log \frac{|\mathcal{F}|}{\deg(c) + 1},$$

and is used to order multiple exact entity matches, not to weight cognitive-map edges.

Each fact has exactly one topic label. Stage 2 applies one shared, entity-anchored canonical-naming contract across sessions, and same-session consolidation selects one representative label for duplicate candidates. On commit, the global topic index is refreshed and semantically equivalent labels are resolved to the same canonical topic cue. Retrieval then reranks the routed query against this normalized global inventory. Consequently, recurring topics provide one reusable cross-session cue while fact text, source pointers, and evidence remain unchanged. Schema gists supply the higher-level routing labels described below.

D.5 Schema Formation

This subsection expands the schema-formation stage in Section 3.4. The schema layer is a hierarchical gist forest with nodes $\mathcal{G}$. It retains evidence links while organizing related experiences into broader rules and preferences.

After one session is committed, complete-linkage clustering is run only over that session's new fact embeddings. The cosine threshold is $\tau_1 = .85$. A cluster of size at least two becomes one disjoint daytime engram. Its daytime memory mark is a pending record containing a temporary identifier, member fact identifiers, cohesion, and source-session identifier, where cohesion is the minimum pairwise cosine in the cluster. It has no gist text, gist embedding, or permanent gist identifier. A singleton remains an orphan. Cues may recur across engrams because engram membership partitions facts, not cue nodes.

At schema formation, each pending engram is first audited for a coherent same-event core. If fewer than two members survive, the group dissolves and its facts remain orphans. Otherwise the synthesis prompt produces a basic gist, its structured answer-time fields, and an embedding over topic, routing cues, gist text, and answer effect. The basic gist keeps all accepted facts as children and unions their entity and evidence links. For uniform forest traversal, a singleton becomes the special orphan basic gist described in the main paper. Operationally, its evidence-bearing payload is the source fact itself, so no additional text generation is required before it participates as a level-0 root. A later accepted higher-level grouping can attach that evidence-bearing root under its generated parent.

Table 8: Memory-construction and schema parameters.

Operation	Setting
Same-session candidate merge	$0.92/0.97$
Daytime engram complete linkage	$\tau_1 = .85$
Higher-root complete linkage	$\tau = .80$
LoCoMo schema interval	10 sessions
PersonaMem-v2 schema interval	5 sessions

Higher-level formation considers the current forest roots, including the special orphan basic-gist representations of singleton facts. Complete linkage at $\tau = .80$ proposes groups. The grouping audit may split or reject a proposal, and the synthesis prompt may decline a group that lacks a shared answer-time consequence. An accepted group receives a new parent at

$$\text{level}(g) = 1 + \max_{u \in \text{child}(g)} \text{level}(u).$$

Every child has at most one parent, and a schema pass raises the accepted roots by only one level. LoCoMo construction runs this pass every ten sessions; PersonaMem-v2 runs it every five sessions and performs a pre-query catch-up whenever earlier sessions have not yet been incorporated.

Cue maintenance and update constraints. The same stage maintains the representations used for cue matching. Entity profiles summarize the fact subgraphs connected to each entity, and topic names are normalized across sessions. These representations support query matching and cue-seed initialization while leaving stored fact text unchanged. Corrections, preference changes, and forgetting requests are preserved as source-grounded facts, and their answer-time consequences are represented by constraint gists. When retrieved, these gists suppress superseded or revoked information in the answer context rather than deleting its source facts.

Retrieval role of the hierarchy. Basic gists link to their supporting facts, and higher-level gists link to their children. Retrieval makes these links bidirectional in the joint memory–schema graph. They allow a higher-level gist to receive PPR activation from multiple descendants, so a relevant abstraction can accumulate more stationary mass than any individual supporting gist. The child links retain the path to the underlying evidence. Gist scores are read from the second PPR pass and selected independently of the first-pass fact scores, so gist constraints do not displace episodic evidence in the final context.

Table 8 collects the fixed consolidation and schema schedules used in the reported runs.

D.6 Update and Forgetting Rules

When a completed session is committed, a newly observed correction, preference change, forgetting request, or privacy instruction is appended as a source-grounded fact. For the current PersonaMem-v2 query, the ongoing conversation remains working context until the answer has been produced. Schema synthesis maps the committed instruction's answer-time consequence to a structured constraint gist. The serialized constraint instructs the answer model to reject superseded options or avoid repeating sensitive values; profile evidence states when a durable preference is applicable, and episode context remains background rather than a global user attribute. A higher-level constraint is accepted only when its children share the same consequence and includes a validity boundary.

At retrieval, constraint gists are placed in the compact schema block before the supporting facts. The answer prompt applies a newer update over an older preference. The evaluated mechanism retains the older fact for provenance and temporal reasoning while changing its answer-time use through the retrieved constraint. For a forgetting or privacy instruction in the current PersonaMem-v2 working context, the same rule is also represented as a temporary hard constraint for that answer and is written to long-term memory only after the answer, together with the rest of the current session.

Table 9: Retrieval settings used by the reported configurations. Iteration counts are maxima; convergence can stop earlier at the stated L_1 tolerance.

Quantity	**LoCoMo**	**PersonaMem-v2**
Flat fact recall	BM25 10 + dense 10	BM25 10 + dense 10; final fact list 10
Entity/topic/fact seed caps	15 / 5 / 3	10 / 10 / 10
Episode recall and seed cap	BM25 8 + dense 8 $\rightarrow$ 2	BM25 5 + dense 5 $\rightarrow$ 5
Fact/episode mass multipliers	1.5 / 2.0	1.5 / 2.0
First-pass PPR	$\alpha = .20$, 30 iter., 10^{-6}	$\alpha = .15$, 30 iter., 10^{-6}
PPR fact width / final facts	20 / 10	10 / 10
Second-pass PPR	$\alpha = .20$, 50 iter., 10^{-8}	$\alpha = .20$, 50 iter., 10^{-8}
Second-pass lexical seed caps	16 cues / 10 facts / 8 gists	16 cues / 10 facts / 8 gists
Reported schema output	none in main LoCoMo rows; Top-2 in update diagnostic	Top-2

E Retrieval Implementation

E.1 Candidate Generation and Reranking

The retained flat fact candidate path has exactly two components: BM25 and dense embedding similarity. BM25 tokenizes the query and each document formed from fact text plus its evidence context. Dense retrieval ranks the same facts by cosine similarity between the query embedding and the stored fact embedding. The two lists are unioned by fact identifier and reranked with `BAAI/bge-reranker-v2-m3` on the query and full candidate text. No temporal, entity, or topic route is an additional flat fact retriever in the reported system.

Entity, topic, and episode representations supply typed graph seeds. An episode candidate document contains timestamp, summary, and raw dialogue. Its BM25 and dense Top-K lists are unioned and reranked against the episode summary. Entity cues use exact name/alias matching followed by semantic profile fill; topic cues are matched through their stored representations. All use the same pretrained reranker without task-specific fine-tuning.

E.2 Query Seeds

Let S_E, S_T, S_F, S_S denote the selected entity, topic, flat-fact, and episode seeds. Within the entity, topic, and fact families, reranker scores are divided by the largest positive score in that family; a zero-score family receives uniform unit weights. The fact family is then multiplied by $\lambda_F = 1.5$. Episode seeds retain their reranker relevance scores and are multiplied by $\lambda_S = 2.0$. After deduplication, all surviving weights are jointly normalized:

$$p_v(x) = \frac{\lambda_{t(v)}\, \widehat{s}_v(x)}{\sum_{u \in S_E \cup S_T \cup S_F \cup S_S} \lambda_{t(u)}\, \widehat{s}_u(x)}.$$

The multiplier is one for entity and topic cues. An empty family contributes no mass; if all families are empty, the query returns no graph result rather than creating an unrelated seed.

Exact entity matches are ordered by entity IDF; semantic profile matches fill remaining slots. Topic seeds map to the exact stored topic-cue identifiers. Flat-fact seeds are taken only after hybrid candidate reranking. Episode seeds are source-session cue nodes rather than raw dialogue inserted directly into the fact ranking.

E.3 Personalized PageRank

For either graph, the implementation initializes $\mathbf{r}^{(0)} = \mathbf{p}$ and iterates

$$\mathbf{r}^{(t+1)} = \alpha \mathbf{p} + (1 - \alpha)\mathbf{P}^\top \mathbf{r}^{(t)}$$

until the maximum iteration count or $\|\mathbf{r}^{(t+1)} - \mathbf{r}^{(t)}\|_1 < \varepsilon$. In the standard first pass, $\mathbf{P}$ is the degree-normalized adjacency of the Bipartite Cognitive Map and only fact-node scores are read. This iteration is equivalent to the coupled fact–cue recurrence in the main paper; the operator $\mathbf{T}_F$ is therefore an interpretation of two-step propagation, not an extra retrieval stage. For the controlled LoCoMo protocol, the evaluation graph additionally links each fact to every source-session output node identified by its evidence pointers, using unit undirected edges. The same PPR assigns stationary mass to these session nodes directly, and the three highest-scoring nodes are returned as raw sessions. This graph readout is the deterministic fact-to-source-session mapping referred to in the main paper; no post-hoc sum or maximum over fact scores is used.

For LoCoMo method-native retrieval, the reranked flat Top-10 and PPR Top-20 are unioned and reranked once more to produce the final ten facts. In the controlled protocol, the session-node readout above produces the three raw sessions. PersonaMem-v2 reads the ten highest-scoring causal facts directly from the first-pass ranking.

The second pass augments the cognitive map with bidirectional fact–basic-gist and child–parent gist edges, each of unit weight. An optional category-cue–gist edge has weight 0.45. Let $L(a, b) = |c(a) \cap c(b)|/\sqrt{|c(a)|\,|c(b)|}$ be count-overlap similarity after lowercase tokenization, punctuation removal, and stop-token filtering. For query q and route ρ, cue seeds score $0.70L(q, v)+0.30L(\rho, v)$, whereas fact and gist seeds score $0.55L(q, v)+0.45L(\rho, v)$. The route is the query for LoCoMo and the query plus current working context for PersonaMem-v2. The pass retains at most 16 cue, 10 fact, and 8 gist seeds; an explicit update or privacy trigger can seed its category cue with weight 3.0. All retained weights are jointly normalized into one restart vector, after which PPR uses the second-pass parameters in Table 9. Only gist-node scores are read. Facts and gists are truncated independently, so a gist never displaces an evidence-bearing fact merely because it has a high joint-graph score.

Retrieval pseudocode

(1) Form BM25 and embedding fact candidates; deduplicate and rerank. (2) Match entity, topic, and episode cues; combine them with reranked fact anchors and normalize the restart vector. (3) Run PPR on the Bipartite Cognitive Map and select facts; in controlled LoCoMo, first attach the unit fact–session output edges and read the Top-3 session nodes directly. (4) When schema retrieval is enabled, add direct gist seeds, run a separate PPR on the joint memory–schema graph, and select gists. (5) Serialize the compact gist constraint block followed by facts; controlled LoCoMo serializes the three raw sessions selected in step (3).

PersonaMem-v2 enforces causality before both passes. A fact is eligible only if its source session precedes the current query, and a gist is eligible only if all supporting descendants satisfy the same boundary. The current session is encoded after its query has been answered.

F Ablation Construction

F.1 Graph Controls

All graph controls use the GPT-4o-mini controlled Top-3 LoCoMo protocol and rank the same raw-session output unit. Table 10 defines the structure changed by each arm.

Let $B_{ET} \in \{0, 1\}^{|\mathcal{F}|\times|\mathcal{C}_E\cup\mathcal{C}_T|}$ be fact incidence on entity and topic cues, let B_{FC} include entity, topic, and episode cues, and let $R_{FS} \in \{0, 1\}^{|\mathcal{F}|\times|\mathcal{S}|}$ be fact–source-session incidence. With stored fact embedding z_i, the reported fact-only control uses, for $i \neq j$,

$$A^{\text{fact}}_{ij} = \mathbf{1}\big[(B_{ET}B_{ET}^\top)_{ij} > 0\big] \max\{0, \cos(z_i, z_j)\},$$

sets the diagonal to zero, and retains the unit fact–session block R_{FS}. The cue-only control uses

$$A_{CC} = B_{FC}^\top B_{FC} - \operatorname{diag}(B_{FC}^\top B_{FC}),$$
$$A_{SC} = R_{FS}^\top B_{FC},$$

where entries count shared facts; its adjacency contains A_{CC} and the symmetric session–cue blocks A_{SC} and $A_{SC}^\top$, with no session–session block.

The full, fact-only, and cue-only rows use the same first-pass seed weights, $\alpha = .20$, at most 30 iterations, and 10^{-6} convergence tolerance. Each projected graph keeps only seeds whose node type

Table 10: Structural controls used for the cognitive-map ablation. The projected graphs are evaluation controls and are not additional structures stored by CORMEM.

Variant	Nodes	Edges and weights	Restart and readout
CORMEM	Facts, entity/topic/episode cues, and session output nodes	Binary fact–cue incidences and unit fact–session output edges	Typed cue and fact seeds; rank session nodes directly
Fact-only	Facts and session output nodes	Shared entity/topic topology, cosine-weighted fact–fact edges, and unit fact–session edges	Retain fact seeds; rank session nodes directly
Cue-only	Entity/topic/episode cues and session output nodes	Shared-fact cue–cue weights and shared-fact session–cue weights	Retain cue seeds; rank session nodes directly
Flat (session control)	Raw sessions	No propagation; dense session ranking over speaker-tagged dialogue	Direct query similarity; return Top-3 sessions

exists in that graph, renormalizes the surviving restart weights, and ranks the explicit session output nodes directly. The projections therefore collapse one side of the bipartite structure while preserving the query anchoring, PPR budget, and session readout. The Flat session control removes propagation altogether and ranks raw sessions directly; it is distinct from the hybrid BM25–embedding flat fact-candidate path in Appendix E. These definitions correspond exactly to the four reported main-paper structural-ablation rows.

F.2 SCHEMA AND REWRITE CONTROLS

The PersonaMem-v2 schema comparison freezes the causal first-pass fact ranking. The *w/o schema* arm serializes its Top-10 facts. The *w/ schema* arm serializes the same Top-10 facts and the Top-2 causal gists selected by the joint-graph pass. The *Top-12 facts* budget control takes the next two facts from the same frozen ranking and adds no gist. Thus the latter comparison tests whether the schema result can be explained by two additional facts.

The *Rewrite* arm replaces separate schema formation with the causal operations in Appendix K.4. It retrieves ten active rewritten facts through the same compact multiple-choice interface. In LoCoMo, the schema/Rewrite table is a diagnostic update comparison; the main LoCoMo retrieval rows remain fact-only outputs with no schema text. PFR and NFR in Appendix H compare each update mechanism with the corresponding no-schema prediction on the same queries.

G BASELINE IMPLEMENTATIONS AND EVALUATION BUDGETS

We evaluate the comparison methods through one protocol that preserves each method's native memory unit. The protocol standardizes dataset ordering, causal visibility, returned-unit budget, and answer formatting; they do not translate a baseline memory into CORMEM facts. Table 11 gives the evaluated interface.

The controlled LoCoMo comparison includes NaiveRAG, HippoRAG 2, and CORMEM; each returns exactly three raw sessions. FullText is an answer-only reference and has no Recall@3. The method-native LoCoMo comparison uses a ten-unit budget for each retrieval method, with Mem0's ten split evenly between the two speakers; FullText retains the complete history. PersonaMem-v2 uses ten method-native units, except HippoRAG 2, which returns three sessions; FullText retains the complete visible history. Queries are processed in timestamp order: the answer sees only prior long-term memory and the current working context, after which the current session is inserted.

Table 11 contains the complete baseline set and the common returned-unit budgets used by the submitted comparisons. Budgets are applied at each method's serialized native-unit boundary; no baseline memory is translated into a CORMEM fact for evaluation. All PersonaMem-v2 entries in

Table 11: Baseline adapters and returned units. MemoryBank denotes the online session-embedding implementation used by the evaluation adapter.

Method	Native returned unit	LoCoMo budget	PersonaMem-v2 budget
FullText	Serialized dialogue history	Complete visible history	Complete visible history
NaiveRAG	Dense-ranked dialogue units; raw sessions in the controlled protocol	Top-3 controlled; Top-10 native	Top-10 sessions
MemoryBank	Session-level memory summaries/embeddings	Top-10	Top-10 memories
HippoRAG 2	PPR-ranked source passages/sessions	Top-3 controlled; Top-10 native	Top-3 sessions
A-MEM	Agentic memory notes	Top-10	Top-10 memories
Mem0	Attributed speaker memories	Top-5 per speaker (at most 10)	Top-10 memories
MemoryOS	Hierarchical context bundles	Top-10	Top-10 memories
LightMem	Compressed long-term memories	Top-10	Top-10 memories

these tables refer to the same formal 100-person, 2,974-query archive used for the submitted main result.

For LoCoMo, the shared prompt in Appendix K.3 changes only the serialized retrieved-memory field. Each adapter renders a ranked unit with its available time, speaker, and provenance metadata, without adding claims absent from the native record. PersonaMem-v2 uses the same current dialogue, question, multiple-choice options, causal boundary, and one-letter output contract for all methods; only the native long-term-memory block and its method-specific format differ.

H Evaluation Protocols and Statistics

H.1 Datasets, Coverage, and Aggregation

LoCoMo contains ten multi-session conversations. We evaluate the 1,540 answerable Category 1–4 questions: 282 multi-hop, 321 temporal, 96 open-domain, and 841 single-hop. The 446 adversarial questions are excluded from F1 and BLEU-1, following the paper's stated scope. Gold-session Recall@3 uses the 1,536 evaluated questions for which a gold source session is defined. Overall F1 and BLEU-1 are query-micro means over the 1,540 questions, rather than an unweighted macro-average of the four category means.

Answers are normalized by lowercasing, removing punctuation and the tokens *a*, *an*, *the*, and *and*, and collapsing whitespace. F1 is Porter-stemmed token overlap. For Category 1, prediction and reference are split at commas; each reference segment takes its best prediction-segment F1 before averaging. For Category 3, scoring uses only the reference text before its first semicolon. BLEU-1 is sentence BLEU with unigram weights and method-1 smoothing over the normalized tokens. The main LoCoMo answer scores average three generations from the same frozen retrieval output, whereas Recall@3 is retrieval-only.

PersonaMem-v2 contains 200 personas and 5,000 scored queries. Before running the memory methods, we selected the 100 personas with the greatest number of eligible scored queries, breaking a tie by persona identifier. This yields 2,974 queries. Accuracy is the query-micro fraction whose selected option matches the gold option. All methods use the same 2,974 keys and causal order.

Efficiency accounting. Tok./Q averages retrieved memory-content tokens across all queries, using `o200k_base` for LoCoMo and the Qwen tokenizer for PersonaMem-v2. CorMem uses 559.1 and 725.4 tokens per query, respectively.

End-to-end memory construction cost is measured with GPT-4o-mini over ten LoCoMo conversations. LLM calls, input tokens, and output tokens are averaged per conversation across the complete construction pipeline.

H.2 LoCoMo

Controlled Top-3 sessions. NaiveRAG, HippoRAG 2, and CorMem return exactly three raw sessions under the answer prompt in Appendix K.3. CorMem attaches explicit source-session output nodes to the cognitive map and returns the three session nodes with greatest stationary PPR mass. Gold-session Recall@3 is

$$\mathrm{R@3}(q) = \frac{|G(q) \cap R_3(q)|}{|G(q)|},$$

where $G(q)$ is the set of annotated gold sessions and $R_3(q)$ is the returned set. Fixing the unit, budget, and answer interface makes this the controlled evidence-ranking protocol.

Method-native Top-10. Each memory system is configured to return up to ten units in its own representation; a query with no retrieved unit is retained under the same fixed denominator. FullText uses the complete history as an answer-only reference. CorMem returns compact source-grounded facts; schema gists, entity-profile text, and source-dialogue expansion are disabled in the answer context. The two protocols use the same 1,540 questions and metric implementation.

Token-matched Top-10 comparison. To separate answer quality from the amount of retrieved text, we conduct an additional comparison over seven baselines and CorMem. We reuse the backbone-specific Top-10 outputs used by the method-native protocol: no memory is reconstructed and no retrieval is rerun. FullText is excluded because it supplies the complete history rather than a Top-10 memory-unit output. For each question, only the serialized retrieved-memory block is matched to exactly 1,024 `o200k_base` tokens, following the tokenizer used in our efficiency analysis. Longer blocks retain their ranked prefix; shorter blocks receive an all-whitespace suffix, which changes token count without adding a memory claim. The question, answer instructions, and generated output are not included in this budget. Across the 36,960 backbone–method–question records, 36,956 native contexts are non-empty and four are zero-length; the latter remain in the archive and receive the same whitespace-padding rule.

All methods use the same shared answer prompt. We independently answer the matched contexts with GPT-4o-mini, Qwen3.5-27B, and DeepSeek-v4-Flash. Thinking is disabled for Qwen and DeepSeek and omitted for GPT; temperature, maximum output length, and structured response format are not sent for any of the three backbones. CorMem uses only compact fact text and does not expose schemas, entity profiles, or source dialogue. The same completeness rule is applied to every method: a missing or empty generation is scored as an empty answer, preserving the fixed 1,540-question denominator.

Table 12 shows that CorMem obtains 48.12/40.33, 50.39/43.02, and 48.75/41.15 F1/BLEU-1 with GPT, Qwen, and DeepSeek, respectively. Its gains over the strongest baseline are 8.34/8.01, 9.93/7.95, and 4.59/4.02 points. The result is therefore consistent across all three answer backbones. This control removes retrieved-text length as an explanation for the overall difference: every method uses the same 1,024-token memory budget while retaining its native serialization. The frozen answer records, exact contexts, native memory blocks, prompt snapshot, token audits, and metric-recomputation script are listed in Appendix J.

H.3 PersonaMem-v2

The main experiment evaluates the 100 personas with the most scored queries, covering 2,974 of the benchmark's 5,000 queries. All methods follow the same causal online order: an answer can use memory constructed from prior sessions and the current working context, but not the current session as long-term memory. Baselines retain their native retrieval interfaces. CorMem returns ten compact facts and two schema gists; its no-schema control returns ten compact facts. Reported accuracies use query-micro aggregation unless explicitly identified as persona-cluster statistics. The submitted main result is 70.48% query-micro accuracy; the corresponding persona-macro accuracy is 70.42%. Ten facts plus two gists average 749.86 retrieved tokens per query, compared with

Table 12: Token-matched Top-10 LoCoMo results under three answer backbones. The retrieved-memory block is fixed to exactly 1,024 `o200k_base` tokens for every method–question pair. F1 and BLEU-1 (%) cover all 1,540 Category 1–4 questions. Each cell reports F1/BLEU-1. Rankings are computed within each backbone. Bold, underline, and italics denote first, second, and third place.

Method	Multi	Temp.	Open	Single	Avg.
GPT-4o-mini					
NaiveRAG	22.78/13.89	8.83/5.93	11.71/8.15	35.40/30.19	26.07/20.78
MemoryBank	31.87/20.36	30.18/25.52	13.70/9.69	33.20/27.81	31.11/24.84
HippoRAG 2	29.88/19.52	*43.39*/*39.02*	16.39/11.51	43.11/36.42	*39.08*/32.32
A-MEM	26.59/17.66	44.25/39.20	15.49/11.13	37.26/31.39	35.41/29.24
Mem0	*38.38*/*25.86*	19.42/17.14	**21.15**/14.81	*47.64*/*40.34*	38.41/31.26
MemoryOS	22.85/15.75	18.05/15.20	12.04/8.97	33.25/27.90	26.86/21.85
LightMem	**43.66**/28.72	19.23/13.53	*17.28*/*12.48*	48.88/41.17	39.77/*31.34*
CORMEM	40.13/**28.95**	**47.78/42.50**	19.33/**14.95**	**54.21/46.21**	**48.12/40.33**
Qwen3.5-27B					
NaiveRAG	16.28/10.93	5.48/3.23	11.40/9.28	35.83/30.87	24.40/20.11
MemoryBank	31.48/22.59	29.83/25.93	17.52/14.63	32.38/27.75	30.76/25.61
HippoRAG 2	28.16/21.76	43.34/38.08	21.91/18.28	45.60/40.31	40.46/35.07
A-MEM	28.51/20.82	**46.72/41.35**	18.60/15.95	41.87/36.52	38.98/*33.37*
Mem0	40.16/29.07	8.91/7.69	*20.39*/*17.02*	53.13/45.79	39.49/32.99
MemoryOS	19.05/12.22	12.31/9.95	14.97/12.28	30.86/26.82	23.84/19.72
LightMem	*35.30*/*27.22*	25.41/21.83	18.34/15.17	*48.87*/*41.50*	*39.59*/33.14
CORMEM	**40.32/31.32**	*43.22*/*37.67*	**23.36/20.18**	**59.59/51.60**	**50.39/43.02**
DeepSeek-v4-Flash					
NaiveRAG	20.89/13.79	6.85/5.09	18.94/15.98	44.81/38.68	30.91/25.71
MemoryBank	32.86/24.30	27.92/23.61	19.10/15.91	34.22/29.23	31.71/26.32
HippoRAG 2	32.19/25.35	**46.79/41.42**	*24.58*/20.62	46.76/41.33	*42.71*/37.13
A-MEM	28.52/21.09	42.78/37.47	18.59/14.31	39.15/33.70	36.68/30.96
Mem0	*37.03*/*27.29*	8.85/7.21	24.93/21.16	*53.44*/*45.69*	39.36/32.77
MemoryOS	21.42/15.11	9.86/8.64	18.44/15.26	37.33/31.87	27.51/22.92
LightMem	38.47/28.73	28.87/24.34	24.54/*20.66*	54.14/46.08	44.16/*36.78*
CORMEM	**40.91/30.61**	*31.80*/*27.66*	**27.94/24.15**	**60.22/51.78**	**48.75/41.15**

620.43 for ten facts. The token-matched Top-12 fact control reaches 67.65% and does not improve over Top-10 ($p = .868$).

H.4 PAIRED FLIP RATES

For N paired queries, let $s_i^{(v)}$ and $s_i^{(0)}$ denote the score of variant v and w/o schema, and let $\Delta_i = s_i^{(v)} - s_i^{(0)}$. The positive flip rate (PFR) and negative flip rate (NFR) are

$$\mathrm{PFR}(v) = \frac{100}{N} \sum_{i=1}^{N} \mathbf{1}[\Delta_i > 0],$$

$$\mathrm{NFR}(v) = \frac{100}{N} \sum_{i=1}^{N} \mathbf{1}[\Delta_i < 0].$$

For LoCoMo, s_i is the unrounded per-question F1 defined above. For PersonaMem-v2, s_i is the binary correctness indicator, so the two events are incorrect-to-correct and correct-to-incorrect flips. Ties remain in N and enter neither numerator. Every paired comparison retains its frozen key set; an archived missing or unusable prediction receives the corresponding empty or incorrect score rather than being removed from N. On LoCoMo ($N = 1{,}540$), w/ schema has 252 positive and 242 negative flips (PFR/NFR $= 16.36/15.71$), while Rewrite has 289 positive and 374 negative flips ($18.77/24.29$). On PersonaMem-v2 ($N = 2{,}974$), the corresponding PFR/NFR values are $7.50/4.81$ for w/ schema and $5.21/3.90$ for Rewrite, matching the update table in the main paper.

Table 13: Retrieval and structural contrasts in percentage points. The first block compares retrieval-only Recall@3 on 1,536 paired LoCoMo questions; within each backbone its Holm family contains the NaiveRAG and HippoRAG 2 contrasts. The second reports the complete map minus each of the three structural controls, with separate three-row Holm families for F1 and R@3.

Comparison	**Metric**	**Δ [95% CI]**	p_{H}
Retrieval-only comparison with HippoRAG 2			
GPT-4o-mini	R@3	+12.90 [9.97, 16.37]	.004
Qwen3.5-27B	R@3	+18.36 [15.56, 21.23]	.004
DeepSeek-v4-Flash	R@3	+16.66 [15.12, 18.41]	.004
Complete Bipartite Cognitive Map minus structural control			
w/o cue (fact-only)	F1	+0.62 [-0.02, 1.28]	.107
	R@3	+4.04 [3.09, 5.17]	.006
w/o fact (cue-only)	F1	+3.43 [2.41, 4.35]	.006
	R@3	+8.43 [6.57, 10.45]	.006
Flat (session control)	F1	+15.65 [13.78, 17.50]	.006
	R@3	+30.42 [27.68, 33.01]	.006

H.5 Confidence Intervals and Significance Tests

The repetition rule is analysis-specific: main LoCoMo QA first averages three generations per question, retrieval-only analyses use the frozen ranking score, and the token-matched and update analyses use their archived per-question predictions. Paired LoCoMo inference preserves question pairing and resamples the ten conversations as clusters. Confidence intervals use 50,000 cluster-bootstrap replicates for the main QA and retrieval comparisons. Paired p-values use a two-sided exact sign-flip test over all $2^{10} = 1{,}024$ conversation-level assignments. PersonaMem-v2 resamples 100 personas; paired tests use 1,000,000 Monte Carlo sign flips of mean within-persona differences. Intervals are 95%, and p_{H} denotes Holm adjustment. The main QA families contain all CorMem-to-method contrasts within one backbone, output protocol, and metric. Retrieval-only comparisons, structural controls, LoCoMo update comparisons, PersonaMem-v2 update/context-budget comparisons, and PersonaMem-v2 external-baseline comparisons are corrected as separate families over all methods in the corresponding frozen result table.

Table 13 reports the retrieval-only and structural contrasts. Table 14 then collects the main QA, update, context-budget, and flip-rate statistics associated with the submitted claims.

Under the controlled Top-3 protocol, the gold-session Recall@3 margins over HippoRAG 2 are $+12.90$, $+18.36$, and $+16.66$ points for GPT-4o-mini, Qwen3.5-27B, and DeepSeek-v4-Flash, respectively; all three cluster-bootstrap intervals exclude zero and all Holm-adjusted values are $.004$. Within the GPT-4o-mini structural comparison, the complete map exceeds the fact-only, cue-only, and flat controls by 4.04, 8.43, and 30.42 Recall@3 points. These paired retrieval results preserve the main paper's separation between evidence ranking and answer generation.

The confirmatory QA rows likewise retain positive F1 and BLEU-1 differences against the named Top-3 and Top-10 comparators across all three backbones. On PersonaMem-v2, CorMem exceeds Mem0 by 4.37 accuracy points and the schema variant exceeds its no-schema control by 2.69 points, both with $p_{\mathrm{H}} < .001$. The Top-12 fact control changes accuracy by only -0.13 points, while the LoCoMo schema arm has a lower negative flip rate than Rewrite (15.71% versus 24.29%). Together, these rows support the submitted retrieval conclusions, confirm the PersonaMem-v2 schema-over-no-schema gain, and support the stated Top-12 context-budget control; the LoCoMo flip-rate comparison remains diagnostic.

I Additional Results and Diagnostics

I.1 Bipartite-Structure Diagnostic with HippoRAG 2

Controlled comparison. This experiment isolates the representation of HippoRAG 2's OpenIE evidence. The converted retriever is built from the exact artifacts used by the native run, rather than from a new extraction pass. The two variants therefore share the same ten LoCoMo stores, session texts and identifiers, frozen OpenIE triples, phrase identifiers, 1,540 Category 1–4 questions, gold-

Table 14: Complete confirmatory statistics associated with the main-paper claims and update controls. Panel a reports the named main QA comparator; Holm families additionally include every method in the corresponding frozen backbone–protocol–metric result table. Panel b shows all rows of the stated external-baseline, LoCoMo-update, and PersonaMem-v2 update/context-budget families. Panel c gives positive/negative unrounded question-level F1 flips. Differences are percentage points.

a. Paired LoCoMo QA comparisons

Setting	Comparator	ΔF1 [CI]; p/p_{H}	ΔB1 [CI]; p/p_{H}
GPT, Top-3	HippoRAG 2	+10.75 [9.66, 11.87]; .002/.006	+10.60 [9.54, 11.64]; .002/.006
GPT, Top-10	A-MEM	+8.40 [5.74, 10.67]; .004/.014	+7.81 [5.31, 10.04]; .004/.014
Qwen, Top-3	HippoRAG 2	+14.25 [11.62, 16.89]; .002/.006	+12.09 [10.07, 14.14]; .002/.006
Qwen, Top-10	A-MEM	+6.05 [4.72, 7.34]; .002/.014	+4.93 [3.79, 6.12]; .002/.014
DeepSeek, Top-3	HippoRAG 2	+17.02 [14.64, 19.62]; .002/.006	+15.13 [12.75, 17.82]; .002/.006
DeepSeek, Top-10	A-MEM	+5.07 [3.03, 7.01]; .006/.018	+3.93 [1.95, 5.73]; .012/.029

b. Personalization, update, and context-budget comparisons

Contrast	Δ [95% CI]	p/p_{H}
PersonaMem-v2 external baseline (accuracy)		
CorMem − Mem0	+4.37 [2.90, 5.85]	$< .001/ < .001$
LoCoMo update comparison (F1)		
w/ schema − w/o schema	+0.45 [-0.41, 1.44]	.436/.436
Rewrite − w/o schema	-2.29 [-3.36, -1.29]	.004/.012
w/ schema − Rewrite	+2.75 [1.56, 3.90]	.004/.012
PersonaMem-v2 update and context budget (accuracy)		
w/ schema − w/o schema	+2.69 [1.36, 4.01]	$< .001/ < .001$
Rewrite − w/o schema	+1.31 [0.34, 2.28]	.008/.024
w/ schema − Rewrite	+1.38 [0.07, 2.67]	.044/.088
Top-12 − Top-10 facts	-0.13 [-1.00, 0.74]	.868/.868

c. LoCoMo paired flip counts and rates (%)

Category (N)	w/ schema: +/-; PFR/NFR	Rewrite: +/-; PFR/NFR
Overall (1540)	252/242; 16.36/15.71	289/374; 18.77/24.29
Multi-hop (282)	39/46; 13.83/16.31	72/59; 25.53/20.92
Temporal (321)	39/29; 12.15/9.03	32/53; 9.97/16.51
Open-domain (96)	14/18; 14.58/18.75	21/24; 21.88/25.00
Single-hop (841)	160/149; 19.02/17.72	164/238; 19.50/28.30

evidence mapping, and Top-3 output budget. The stored native HippoRAG 2 retrievals are reused without regeneration.

All models and numerical retrieval settings are also held fixed. Both variants use GPT-4o-mini for HippoRAG 2 fact recognition, `text-embedding-3-large` for dense representations, five candidates for fact linking, cosine-weighted phrase-synonym edges above 0.8, a dense-session restart weight of 0.05, and weighted undirected PRPACK PPR with damping 0.5. Sessions are ranked by their stationary PPR probability. CorMem is the frozen no-schema result from the same Top-3 session protocol. We treat the native-to- conversion and conversion-to-CorMem differences as two pre-specified adjacent comparisons.

Graph conversion. Native HippoRAG 2 stores two graph node types: phrase nodes and session nodes. For an OpenIE triple (s, r, o), s and o become phrase nodes; r remains in the separately stored triple used for query–fact linking and is not a graph node. The graph adds a weighted s–o co-occurrence edge, whose weight grows when the pair is extracted repeatedly, and unit-weight

Table 15: Actual minimal rewrite from `conv-42`, session 2 (`fact-occ-0030`). The rewrite adds only the omitted subject and copula (italicized); it introduces no new claim.

Form	Stored content
OpenIE triple	(Joanna; wants to get; pets if not allergic)
Fact sentence	Joanna wants to get pets if *she is* not allergic.

links from the distinct subject/object phrases in a session to that session node. It also adds cosine-weighted phrase–phrase synonym edges. Thus, in the native graph, sessions connect directly to phrases and an individual triple is not an explicit graph node.

The conversion is an occurrence-level edge replacement. We retain every native phrase node and every native phrase–phrase synonym edge, including its cosine weight and 0.8 threshold. We remove the native subject–object co-occurrence edges and direct phrase–session edges. For each extracted triple occurrence in session d, we instead create a distinct fact node $f_{t,d}$ and add three unit-weight edges: s–$f_{t,d}$, o–$f_{t,d}$, and $f_{t,d}$–d. If the same normalized triple appears in two sessions, it produces two fact nodes, so each occurrence preserves its own source path. We add no fact–fact edge. The resulting graph is therefore a phrase–fact–session bipartite backbone augmented only by HippoRAG 2's unchanged phrase-synonym edges; it is not a mathematically pure bipartite graph.

Each fact node stores a minimally completed sentence because the raw OpenIE tuple is often not independently readable. GPT-4o-mini receives one source session and all triples extracted from that session, then returns one sentence per triple. The prompt permits unambiguous grammatical completion but forbids new events, causes, emotions, dates, relationships, cross-session information, or future information. Table 15 reports an actual occurrence from `conv-42`; the 4,802 resulting fact sentences average 5.61 words.

Retrieval and fixed parameters. Query processing follows HippoRAG 2's native sequence in both variants. First, query–fact cosine similarity is min–max normalized and the top five fact candidates are sent to the same GPT-4o-mini fact-recognition prompt. Native HippoRAG 2 embeds the frozen OpenIE tuples. The conversion embeds the minimally completed sentences because they are the new fact-node contents, but presents the corresponding original triples to the recognition prompt. For each recognized triple, the normalized fact score is divided by the number of source sessions containing each subject/object phrase. Scores from multiple recognized triples are averaged per phrase, and only the five highest-weighted phrases enter the restart vector. This is the native degree correction and Top-5 phrase budget.

The dense restart channel is unchanged: every session receives its min–max-normalized query similarity multiplied by 0.05. The conversion adds a restart channel only for the inserted fact nodes. A recognized triple's score is divided equally among its source-specific fact occurrences, preventing a triple repeated across sessions from receiving duplicated total restart mass. The phrase, fact, and dense-session weights are summed into one reset vector; weighted undirected PRPACK PPR is then run with damping 0.5. Both variants return the three session nodes with the highest stationary probabilities.

Results and interpretation. Under these controls, the conversion raises overall Recall@3 from 73.93 to 77.39, a gain of 3.46 points (Table 16). Recall increases in all four question categories and in all ten memory stores. The exact conversation-cluster test gives raw $p = .002$ and Holm-adjusted $p_{\mathrm{H}} = .004$, and the cluster-bootstrap interval excludes zero. Because the LLM, embedding model, candidate budget, synonym graph, dense seed, PPR, and output budget are identical, these factors cannot explain the difference. The measured change therefore belongs to the controlled conversion: the minimally completed fact-node text, rewired incidence paths, and associated fact-node restart channel. It is retrieval-only and does not depend on answer generation. The conversion recovers 26.8% of the 12.90-point gap between native HippoRAG 2 and CorMem. CorMem remains 9.45 points above the conversion, with a 95% cluster-bootstrap interval of $[7.83, 11.52]$ and Holm-adjusted $p_{\mathrm{H}} = .004$.

The conversion deliberately transfers only one structural principle. A phrase repeated across source-specific fact occurrences now creates distinct phrase–fact incidence paths, making part of the corresponding co-occurrence structure traversable under query-conditioned PPR. This recovers some of the benefit of representing latent relations through shared incidence paths, but it does not reproduce

Table 16: Detailed Recall@3 (%) for the HippoRAG 2 conversion on 1,536 paired LoCoMo questions. Overall contrasts use ten conversation clusters, exact two-sided sign-flip tests, and 95% cluster-bootstrap intervals. Both raw $p = .002$, with Holm correction over the two contrasts. Differences use unrounded scores.

Category	**HippoRAG 2**	**Conversion**	**CorMem**
Multi-hop	55.36	57.25	62.02
Temporal	78.56	82.35	89.77
Open-domain	50.14	55.15	56.56
Single-hop	80.99	84.68	97.34
Overall	73.93	77.39	86.83
Overall contrast	**Δ (points)**	**95% CI**	$p_{\mathbf{H}}$
Conversion − HippoRAG 2	+3.46	$[2.19, 5.25]$	.004
CorMem − Conversion	+9.45	$[7.83, 11.52]$	.004

CorMem. CorMem assigns three typed cue families (entity, topic, and episode) during source-grounded fact encoding, preserves their cue–fact incidences, and accumulates evidence across multiple semantically distinct shared-cue paths. The HippoRAG 2 conversion remains limited to subject/object phrases from OpenIE, retains native synonym edges, and lacks CorMem's typed topic and episode access paths. CorMem consequently retains a 9.45-point advantage.

The conversion and the main-paper ablation provide complementary evidence for the same mechanism. Projecting CorMem's complete bipartite map to fact-only or cue-only graphs reduces Recall@3 from 86.83 to 82.79 and 78.40, respectively, whereas adding fact-mediated incidence paths to the controlled HippoRAG 2 representation raises Recall@3 from 73.93 to 77.39. Both diagnostics support the role of the complete cue–fact topology reported in the main paper.

I.2 Relational Organization Analysis

Inputs. The analysis reuses the frozen GPT-4o-mini LoCoMo retrievals of the main paper and runs no new retrieval, answer generation, or judgment. Session-unit methods contribute their Top-3 sessions (CorMem session readout, HippoRAG 2, and session-level NaiveRAG). Native-unit methods contribute the source sessions of their Top-10 units: CorMem facts through their source-session identifiers and evidence pointers, HippoRAG 2 passages through their session metadata, Mem0 and MemoryBank through stored session identifiers, and NaiveRAG, A-MEM, LightMem, and MemoryOS pages through their stored session timestamps. MemoryOS knowledge entries carry no session provenance and do not contribute. As a consistency check, the reconstructed Top-3 gold-session recall of CorMem (session), HippoRAG 2, and session-level NaiveRAG is 86.86, 73.92, and 56.48, respectively. These provenance-reconstructed recalls describe the evidence-eligible diagnostic subset; they do not replace the main paper's reported Recall@3 values (86.83, 73.93, and 56.41, respectively).

Co-evidence reference. For conversation c, let Q_c contain the Category 1–4 questions whose resolvable evidence spans at least two sessions, and let S_q be the deduplicated gold sessions of question q. With $Z_{qi} = \mathbb{1}[i \in S_q]$,

$$C_{ij} = \sum_{q \in Q_c} Z_{qi} Z_{qj}, \qquad f_i = \sum_{q \in Q_c} Z_{qi}, \qquad G_{ij} = \frac{C_{ij}}{\sqrt{f_i f_j}} \quad (i \neq j), \tag{6}$$

on the evaluated sessions $V_c = \{i : f_i > 0\}$. Zero entries denote pairs never annotated as joint evidence.

Retrieval-induced association. For method m, $M_m[q, i] = 1$ when session i is among the sessions returned for question q, over all 1,536 Category 1–4 questions with gold evidence. The association between sessions i and j is the Pearson (ϕ) correlation of columns $M_m[:, i]$ and $M_m[:, j]$; it is set to zero when either column is constant. Two sessions are associated when the method tends to return them for the same questions and not otherwise.

Alignment and statistics. Per conversation, $\rho_{m,c} = \mathrm{Spearman}(\mathrm{upper}(G_c), \mathrm{upper}(A_{m,c}))$ over all pairs in V_c, and AUROC measures how well $A_{m,c}$ ranks pairs with $G_{ij} > 0$ above pairs with

$G_{ij} = 0$. Scores are averaged with equal conversation weight. Intervals use 2,000 conversation-cluster bootstrap resamples, and p-values use a two-sided exact paired sign-flip test over the ten conversations (smallest attainable $p = 2/1024$). These exploratory pairwise p-values are unadjusted for multiple comparisons.

Relational geometry. For G and each A_m, off-diagonal entries on V_c are converted to percentile ranks with averaged ties, and $D_{ij} = 1 - \mathrm{rank}(A_{ij})$. Classical MDS (Torgerson, 1952) embeds D in two dimensions. Each method map is aligned to the gold map by Procrustes analysis (Gower, 1975), which allows only translation, rotation or reflection, and uniform scaling. The disparity d_P is the residual sum of squares between the standardized configurations (0 for identical geometry). The displayed conversation is chosen by a fixed rule: among conversations where CORMEM has the lowest d_P of the eight displayed methods, the one with the largest ρ margin over the displayed baselines (conv-47). Averaged over conversations, d_P is lowest for LightMem (0.814) and 0.826 for CORMEM (Table 21); the geometry panel is illustrative, and quantitative comparisons rest on ρ.

Matched Top-3 session readout. Native protocols return unequal numbers of sessions. Under a matched Top-3 session budget, CORMEM remains among the strongest methods ($\rho = .223$), statistically tied with LightMem (.222) and above the remaining baselines ($p \leq .047$, Table 18). Short lists are extended by first-pass PPR for CORMEM (499 questions) and by each baseline's own scorer (Mem0 228, LightMem 51, MemoryOS 307, NAIVERAG 7).

Cue-type ablation. First-pass PPR is recomputed from saved seeds after removing selected cue families ($\alpha = 0.2$). Table 19 reports session Top-3 and fact Top-10. No cue halves alignment on both readouts ($p < .035$). w/o Episode does not, as episode cues link only intra-session facts.

Table 17: Relational organization over ten LoCoMo conversations (GPT-4o-mini), native main-table protocols. Values are mean [95% CI] (unadjusted p versus CORMEM). Recall is gold-session recall of the returned unit set. U10: Top-10 native units; S3: Top-3 sessions.

Method	Unit	Recall	Spearman ρ	AUROC
CORMEM	U10	90.24	0.248 [0.18, 0.32]	0.670 [0.62, 0.73]
CORMEM (session)	S3	86.86	0.174 [0.12, 0.22] (.020)	0.615 [0.58, 0.65] (.018)
HippoRAG 2	S3	73.92	0.161 [0.09, 0.23] (.002)	0.604 [0.56, 0.65] (.004)
NAIVERAG (session)	S3	56.48	0.131 [0.10, 0.17] (.006)	0.587 [0.56, 0.62] (.012)
A-MEM	U10	91.41	0.163 [0.11, 0.22] (.020)	0.609 [0.57, 0.65] (.014)
Mem0	U10	84.60	0.182 [0.11, 0.24] (.039)	0.616 [0.57, 0.65] (.029)
LightMem	U10	90.27	0.218 [0.16, 0.28] (.266)	0.647 [0.61, 0.69] (.244)
MemoryOS	U10	54.06	0.118 [0.03, 0.19] (.006)	0.583 [0.53, 0.63] (.010)
MemoryBank	U10	83.56	0.122 [0.08, 0.16] (.002)	0.579 [0.55, 0.61] (.006)
NAIVERAG	U10	69.08	0.122 [0.05, 0.19] (.043)	0.576 [0.53, 0.62] (.025)

Table 18: Matched Top-3 session readout. Every method is aggregated to three distinct sessions before A is formed. Ext.: questions whose saved Top-10 list was extended by the method's own scorer. Values are mean [95% CI] (unadjusted p versus CORMEM).

Method	Ext.	Recall	Spearman ρ	AUROC
CORMEM	499	87.25	0.223 [0.18, 0.28]	0.648 [0.61, 0.70]
CORMEM (session)	0	86.81	0.174 [0.12, 0.22] (.047)	0.615 [0.58, 0.65] (.072)
HippoRAG 2	0	73.50	0.161 [0.09, 0.23] (.020)	0.604 [0.56, 0.65] (.027)
NAIVERAG (session)	0	57.05	0.131 [0.10, 0.17] (.002)	0.587 [0.56, 0.62] (.008)
A-MEM	0	68.74	0.077 [0.02, 0.13] (.002)	0.546 [0.50, 0.59] (.002)
Mem0	228	78.56	0.143 [0.08, 0.20] (.008)	0.587 [0.55, 0.62] (.010)
LightMem	51	81.98	0.222 [0.17, 0.28] (.957)	0.645 [0.61, 0.69] (.887)
MemoryOS	307	52.00	0.107 [0.04, 0.17] (.010)	0.574 [0.53, 0.62] (.020)
MemoryBank	0	58.11	0.141 [0.09, 0.19] (.025)	0.590 [0.56, 0.62] (.033)
NAIVERAG	7	54.24	0.082 [0.01, 0.15] (.002)	0.551 [0.52, 0.59] (.002)

Table 19: Cue-type ablation of CorMem PPR, recomputed from saved seeds with no new retrieval. Session Top-3 and fact Top-10 are the two native readouts. Parentheses are unadjusted p versus Full. w/o cue removes every cue node and cue seed.

	Session Top-3		Fact Top-10	
Variant	Spearman ρ	Recall	Spearman ρ	Recall
Full	0.174	86.81	0.215	88.53
w/o Episode	0.181 (.182)	86.03	0.215 (.967)	88.53
w/o Entity	0.188 (.430)	85.70	0.160 (.008)	86.99
w/o Topic	0.176 (.871)	87.07	0.216 (.955)	86.72
w/o cue	0.090 (.035)	83.58	0.106 (.002)	82.11

Table 20: Per-conversation alignment ρ between $\mathrm{upper}(G)$ and $\mathrm{upper}(A)$ (native protocols). Bold marks the best method per conversation. $|V|$: evaluated sessions; nz: non-zero G pairs. Column headers give the numeric suffix of each conversation ID.

Method	26	30	41	42	43	44	47	48	49	50	Mean
CorMem	0.165	0.068	**0.308**	0.238	**0.297**	0.178	**0.517**	**0.220**	0.271	**0.217**	**0.248**
BM (sess.)	-0.033	0.079	0.249	0.236	0.182	0.175	0.246	0.138	0.271	0.198	0.174
HippoRAG 2	0.011	0.047	0.227	0.210	0.252	0.120	0.306	-0.002	0.265	0.170	0.161
Naive (sess.)	0.104	0.094	0.071	0.131	0.071	0.167	0.233	0.064	0.233	0.140	0.131
A-MEM	0.044	0.178	0.212	0.199	0.226	0.140	0.323	0.103	0.170	0.038	0.163
Mem0	0.131	-0.077	0.245	0.283	0.230	0.182	0.236	0.184	0.292	0.115	0.182
LightMem	0.121	0.066	0.266	**0.334**	0.245	**0.192**	0.361	0.175	**0.331**	0.093	0.218
MemoryOS	**0.200**	-0.199	0.125	0.094	0.235	0.157	0.254	0.061	0.197	0.057	0.118
MemBank	0.056	0.049	0.175	0.237	0.120	0.043	0.187	0.112	0.195	0.047	0.122
Naive	0.166	**0.310**	-0.071	0.109	0.135	0.033	0.288	-0.012	0.162	0.098	0.122
$\|V\|$	18	11	32	28	29	27	21	26	25	29	–
nz	43	15	69	118	93	89	34	72	194	60	–

Table 21: Per-conversation Procrustes disparity d_P between each method's MDS map and the gold map (lower is closer). Bold marks the lowest disparity per conversation. Column headers give the numeric suffix of each conversation ID.

Method	26	30	41	42	43	44	47	48	49	50	Mean
CorMem	0.990	0.812	0.894	0.823	0.798	0.854	**0.583**	0.933	0.701	**0.872**	0.826
BM (sess.)	0.938	0.812	**0.778**	0.879	0.870	0.831	0.726	0.878	0.811	0.905	0.843
HippoRAG 2	0.930	0.710	0.907	0.886	0.821	0.958	0.856	0.887	0.925	0.886	0.877
Naive (sess.)	0.958	0.642	0.909	0.974	0.893	0.894	0.809	0.877	0.818	0.876	0.865
A-MEM	0.997	0.778	0.859	0.812	0.885	0.947	0.818	0.878	0.909	0.957	0.884
Mem0	**0.892**	0.857	0.901	0.817	0.810	0.814	0.942	0.831	0.755	0.888	0.851
LightMem	0.953	0.845	0.875	**0.778**	0.792	0.792	0.637	0.874	0.696	0.898	**0.814**
MemoryOS	0.954	0.884	0.891	0.914	**0.721**	**0.573**	0.925	0.898	0.890	0.882	0.853
MemBank	0.998	0.792	0.846	0.918	0.769	0.942	0.950	**0.798**	**0.687**	0.924	0.862
Naive	0.968	**0.629**	0.888	0.916	0.897	0.919	0.916	0.970	0.936	0.880	0.892

I.3 Seed-Family Contribution to Retrieved Sessions

Protocol. We use the frozen GPT-4o-mini run from the controlled Top-3 LoCoMo protocol and retain questions satisfying two conditions. First, CorMem's mean answer F1 over three generations is at least the strongest same-question comparator in that panel (FullText, NaiveRAG, or HippoRAG 2), including ties. Second, CorMem's Top-3 output contains at least one gold session. This produces 1,133 questions. The outcome-based filter is intentional: the analysis asks which restart channels supply PPR mass to correct sessions when CorMem is not worse than the strongest comparator. It is therefore a conditional mechanism audit, not an estimate over the full benchmark and not a causal test of answer F1.

We replay the saved query seeds on the frozen graph without calling an LLM, embedding model, or reranker. The retrieval configuration is unchanged: $\alpha = .20$, at most 30 iterations, tolerance 10^{-6}, and budgets of 15 entity, 5 topic, 2 episode, and 3 flat-fact seeds. The stored query-specific weights are retained, including the 2.0 episode and 1.5 flat-fact channel factors, and are normalized once to form the restart distribution.

Exact contribution decomposition. For the finite recurrence $\mathbf{r}^{(t+1)} = \alpha\mathbf{p} + (1-\alpha)P^\top\mathbf{r}^{(t)}$, we partition $\mathbf{p}$ into entity, topic, episode, and flat-fact components. Linearity then gives an exact decomposition of the PPR score of each retrieved gold session into the four seed families, including all graph walks and the finite-iteration residual. When multiple gold sessions occur in the returned Top-3, we sum their source-specific mass and normalize the four components to 100%. This differs from counting representative paths: every valid walk contributes according to its probability.

Results. All percentages below are conditioned on the advantage-or-tie questions for which CorMem retrieves a gold session, so they describe how effective retrievals are assembled rather than performance on arbitrary questions. Flat-fact seeds provide 55.8% of the PPR mass on these gold sessions, while the three typed cue channels jointly provide 44.2% (Figure 7). Flat facts remain the direct evidence-bearing entry channel, while entity, topic, and episode seeds provide the complementary object, thematic, and interaction-context views defined in the main paper. The decomposition therefore confirms that all four implemented seed families contribute to the scores of retrieved gold sessions.

I.4 Mechanism Examples

We next inspect two frozen LoCoMo queries. The fact-level case uses the same Top-10 graph and hybrid fact-candidate arms, fact store, reranker, and answer prompt. The session-level case uses the controlled Top-3 protocol. A path below is a verified path in the stored graph, but it is not a decomposition of all walks that contribute to the PPR score.

Fact-level graph rescue. For Q1631, "Which places in Canada was Evan visiting in July 2023?", the gold answer is *Banff, Rocky Mountains*. The hybrid fact-candidate arm is occupied by surface-similar but time-mismatched memories, including a future Canadian honeymoon, and retrieves neither gold fact. The graph admits both facts through the location cues. For Banff, a representative verified path is *Banff* → *Evan drove to Banff for skiing* → *Evan skiing experience* → *Evan skied in Banff in July*; the Rocky Mountains fact is reached through its location cue. The two facts enter the merged reranker at ranks 10 and 9, respectively. CorMem then returns the exact two-place answer in all three generations (F1/BLEU-1 $= 100/100$), whereas the strongest memory-unit baseline averages 40.0 F1. This case illustrates the intended role of propagation: it expands the candidate set beyond facts with the highest direct query similarity.

Session-level convergence. Figure 8 traces Q1480, "What is Jolene currently doing in June 2023?" Topic, entity, fact, and episode seeds connect to evidence from session 13, which ranks first after PPR. CorMem achieves 100.0 Recall@3 and 72.7 F1, whereas NaiveRAG and HippoRAG 2 miss the target session. The dialogue excerpt grounds the internship answer and the additional yoga detail, which refers to the previous morning and is absent from the gold answer.

a. Contribution by question type (%)

		Cue contribution				Fact
Split	N	**Entity**	**Topic**	**Episode**	**Total**	**Share**
Overall	**1,133**	**21.0**	**13.5**	**9.7**	**44.2**	**55.8**
Multi-hop	193	22.3	15.1	10.4	47.7	52.3
Temporal	238	18.1	15.1	7.5	40.7	59.3
Open-domain	45	16.5	22.7	6.2	45.3	54.7
Single-hop	657	22.0	11.9	10.5	44.4	55.6

b. Per-question contribution

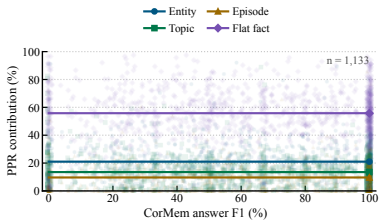

Figure 7: Seed-family contribution to retrieved gold sessions on the conditional LoCoMo subset. a. Category-wise means; each row sums to 100%. b. Exact per-question shares against CorMem answer F1. Horizontal lines denote overall means. A deterministic horizontal jitter separates coincident F1 values only for display; all statistics use the original values.

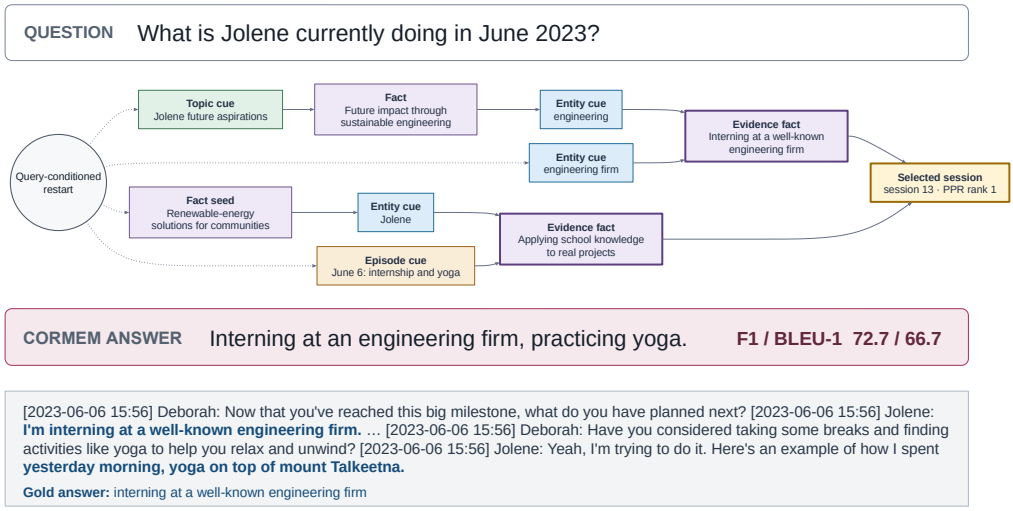

Figure 8: Session-level retrieval example for LoCoMo Q1480. Dashed arrows indicate query-conditioned restart channels, and solid arrows show stored cue–fact incidences or fact-to-session mappings. Representative routes converge on evidence from session 13, ranked first by PPR. The gray panel presents timestamped dialogue excerpts with relevant statements highlighted and the gold answer below.

I.5 PersonaMem-v2 Update Cases

Table 22 shows one constraint-guided flip from the frozen 100-persona Qwen3.5-27B run. Facts and the schema gist are shown separately to make the update mechanism inspectable without changing the source evidence.

In the constraint case, the no-schema and Rewrite answers both retain the update fact but reason that a specific reenactment is distinct from a cultural heritage festival. The schema gist instead preserves the semantic scope of the forget request and selects the only option outside that class. The result supports a narrow claim: a separate gist can operationalize an update even when the current context and stored facts make the superseded choice highly salient.

Table 22: Representative PersonaMem-v2 update case from frozen outputs. The case shows schema-based constraint execution under conflicting current context while the source-grounded fact remains unchanged.

Case	Fact and current-context evidence	Retrieved schema gist	Frozen predictions
Constraint success (persona 439)	The stored fact records Anika's request to forget her preference for cultural heritage festivals. The current translation concerns a historical reenactment, which strongly primes that same attraction type.	Treat the forgotten festival preference as a hard exclusion in future recommendations; retain the separate preference for outdoor events that support local businesses.	Gold: D (food-truck rally). CorMem: D. No schema: C; Rewrite: C. The external baselines choose A or C.

I.6 Hyperparameter Sensitivity

On 240 stratified LoCoMo questions, F1 rises from 54.47 at $K = 10$ to 56.47 at $K = 40$, then falls to 55.34 at $K = 50$. We retain $K = 10$ for cost–quality balance. On 270 retrieval-only questions, Recall@10 peaks at 54.13 for the default $\alpha = .20$; Figure 9 shows both sweeps.

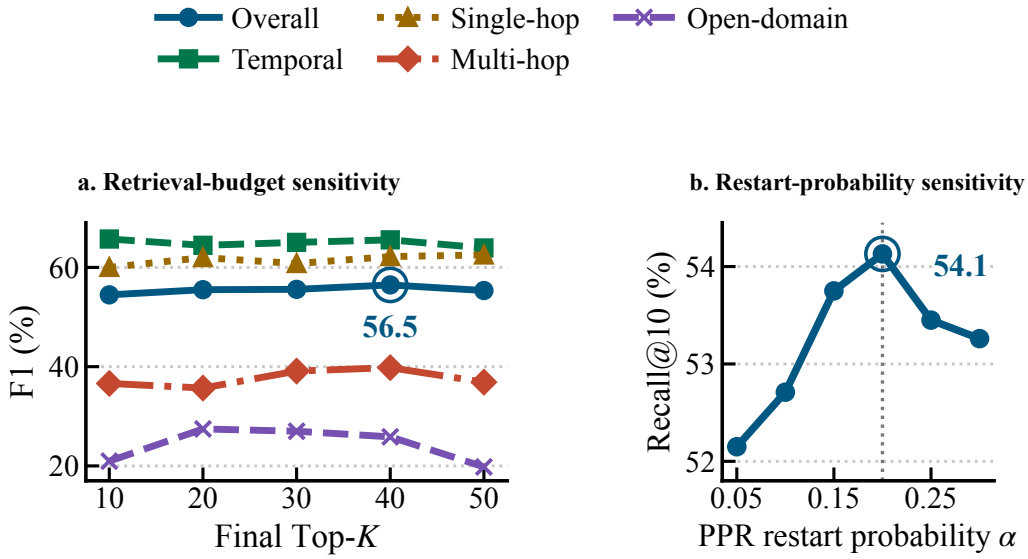

Figure 9: Hyperparameter sensitivity on LoCoMo. Left: F1 versus the final fact budget K. Right: retrieval-only Recall@10 versus the restart probability α.

I.7 Additional Diagnostics

The preceding analyses address candidate coverage, graph propagation, memory updating, and answer generation at distinct levels. Retrieval-only Recall@3 and the seed-family decomposition examine evidence ranking; the fact-level and session-level cases trace concrete paths; and the PersonaMem-v2 case illustrates constraint application. The token-matched study in Appendix H (Table 12) separately controls retrieved-text length. Its archived answer records, exact contexts, prompt snapshot, token audits, and metric-recomputation script are summarized in Appendix J.

I.8 Reported Comparisons with Recent Structured-Memory Systems

Recent work has explored complementary graph and hierarchical representations, including associative event graphs, episodic–semantic activation graphs, event–turn hierarchies, and multi-graph

Table 23: Contextual LoCoMo F1 values under GPT-4o-mini. CORMEM is the method-native Top-10 result reported in the main paper. AnchorMem, Synapse, and MAGMA are their reported Category 1–4 aggregates from independent pipelines. HiGMem is a question-count-weighted summary of its four reported Category 1–4 scores using the counts in Appendix H. No controlled cross-row comparison is implied.

System	Backbone	Avg. F1 (%)
CORMEM (ours)	GPT-4o-mini	52.79
AnchorMem	GPT-4o-mini	49.87
Synapse	GPT-4o-mini	40.50
HiGMem	GPT-4o-mini	40.46
MAGMA	GPT-4o-mini	46.70

memories (Shen et al., 2026; Jiang et al., 2026b; Cao et al., 2026; Jiang et al., 2026a). Table 23 summarizes LoCoMo F1 values when the answer backbone and metric are sufficiently close to ours. Because the external values were produced by independent evaluation pipelines, we treat them as contextual evidence rather than controlled comparisons. The CORMEM row repeats the method-native GPT-4o-mini result from the main paper, whose uniformly rerun comparisons remain the basis for our comparative claims.

J TOKEN-MATCHED REPRODUCIBILITY ARTIFACTS

Appendix H defines the token-matching rule, shared answer setting, fixed denominator, and metrics, and reports the results in Table 12; the shared answer prompt is given in Appendix K.3. Table 24 records only the saved quantities and checks required to audit and recompute that table without a new retrieval or answer-model call.

Table 24: Minimum archived data and computation checks for the token-matched LoCoMo control defined in Appendix H.

Quantity	Value or check
Backbones	3: GPT-4o-mini, Qwen3.5-27B, and DeepSeek-v4-Flash.
Methods	8 per backbone: the seven baselines and CORMEM shown in Table 12.
Questions	1,540 per method: 282 multi-hop, 321 temporal, 96 open-domain, and 841 single-hop.
Result rows	$3 \times 8 \times 1{,}540 = 36{,}960$, each with a stable backbone–method–question identifier.
Context records	36,960 native-context records (36,956 non-empty and four zero-length records retained under the fixed denominator) and 36,960 exact matched contexts.
Token check	Exactly 1,024 `o200k_base` tokens for every row.
Answer rows	36,952 status-`ok` predictions plus eight `content_filtered` empty predictions for one GPT-4o-mini question across all methods; F1 and BLEU-1 retain all 1,540 rows per method.
Archived files	Native and exact contexts, canonical results, token audits, prompt and configuration snapshots, metric code, combined summaries, and a 37-entry package SHA-256 manifest.

Computation. For each backbone, the archived summarizer reads the configuration, exact matched contexts, and canonical results, and joins context and result by `task_id`. It checks the configuration values, exact 12,320-row method/category coverage, the 1,024-token invariant, context–result hashes, and prompt, request, and tokenizer metadata for status-`ok` records. It then recomputes F1 and BLEU-1 and regenerates the JSON, CSV, Markdown, and LaTeX summaries using the category partitions and denominator in Appendix H. Native blocks and token audits remain archived

provenance rather than inputs to this metric-recomputation script; the 37 package checksums are verified separately. No retrieval or answer-model call is repeated.

K Operational Prompt Templates

The prompt templates used by the reported implementation are given below. Each box retains the operative role, rules, run-time input positions, complete output contract, and a format example. Double braces mark fields filled at run time. Every generative construction stage returns a JSON object; entity, topic, and episode retrieval templates are reranker queries rather than additional generative extraction calls.

K.1 Memory Encoding Prompts

The three stages operate on the same timestamped session and preserve source turn identifiers across their outputs.

Prompt K.1 Stage 1: session-wide entity discovery

Task. You are an entity extractor for a temporal personal context graph. Extract graph-useful entities from one timestamped dialogue session. Coverage is important: include speakers and aliases, people, pets, owner-specific groups, organizations, places, jobs and roles, meaningful events and processes, recurring activities, important objects, goals, plans, values, emotions, motivations, opinion targets, and useful visual subjects in `[image: ...]` descriptions.

Rules. Use identifiers `E1,E2,...`; choose the shortest clear canonical name; preserve original names and put supported alternatives in `aliases`. Merge only clear aliases. Keep an object distinct from the activity involving it. Do not output greetings, generic pronouns, temporary dialogue actions, vague words, or a complete fact as an entity. Allowed types are `person`, `pet`, `animal_category`, `group`, `place`, `job_role`, `object`, `activity`, `organization`, `concept`, and `unknown`. Use only the supplied session.

Runtime input.

```
Session timestamp: <ISO time>
<current_dialogue>
{{CURRENT_DIALOGUE}}
</current_dialogue>
```

Output. Return valid JSON only:

```
{
  "entities": [{
    "id": "E1",
    "name": "<canonical entity name>",
    "type": "<allowed type>",
    "aliases": [],
    "evidence": ["<turn_id>"],
    "confidence": "<high|medium|low>"
  }]
}
```

Example.

```
Input:
Session timestamp: 2023-06-09T19:55:00
D3:1 Caroline: I attended an LGBTQ support group last week.

Output:
{"entities":[
  {"id":"E1","name":"Caroline","type":"person",
   "aliases":[],"evidence":["D3:1"],"confidence":"high"},
  {"id":"E2","name":"LGBTQ support group","type":"organization",
   "aliases":[],"evidence":["D3:1"],"confidence":"high"}
]}
```

Prompt K.2 Stage 2: per-entity fact extraction

Task. You are a focused fact extractor for a temporal personal context graph. Emit every memory-worthy atomic fact involving the selected entity, using only the supplied entity inventory and dialogue.

Rules. Every fact must involve `selected_entity` and reuse entity identifiers from `entities`; do not invent identifiers or `new_entities`. Preserve supported attribution, relationship, motivation, constraint, and personal meaning. Merge details only within one core memory, but split distinct topics, retrieval questions, time anchors, plans, motivations, values, and durable relations. Give each fact exactly one concise reusable topic. Treat image descriptions as evidence without inferring beyond them. Normalize real-world time against the session timestamp: *last week* is the previous calendar week; use an approximate range for durations; use `mixed` when a dated span and a present observation co-occur. Mention time is not event time. Use no outside knowledge.

Runtime input.

```
Session timestamp: {{SESSION_TIMESTAMP}}
<selected_entity>
{{SELECTED_ENTITY}}
</selected_entity>
<entities>
{{ENTITIES}}
</entities>
<text>
{{TEXT}}
</text>
```

Output. Return valid JSON only:

```
{
  "selected_entity": {"id":"", "name":""},
  "facts": [{
    "fact": "",
    "topic": "",
    "entities": [],
    "time": {
      "value": [],
      "kind": "<event|state_observed|preference_observed|habit_observed|
               plan_observed|intention_observed|belief_observed|
               motivation_observed|opinion_observed|valid_range|
               approximate_range|mixed|unknown>",
      "confidence": "<high|medium|low>",
      "reason": ""
    },
    "evidence": [],
    "confidence": "<high|medium|low>"
  }]
}
```

Example.

```
Input:
Session timestamp: 2023-06-09T19:55:00
selected_entity = {"id":"E1","name":"Caroline","type":"person"}
entities = [{"id":"E1","name":"Caroline","type":"person"},
            {"id":"E2","name":"LGBTQ support group",
             "type":"organization"}]
D3:1 Caroline: I attended an LGBTQ support group last week.

Output:
{"selected_entity":{"id":"E1","name":"Caroline"},"facts":[
{"fact":"Caroline attended an LGBTQ support group during the
previous calendar week.","topic":"Caroline support group visit",
"entities":["E1","E2"],
"time":{"value":["2023-05-29","2023-06-04"],"kind":"event",
"confidence":"medium","reason":"Previous calendar week relative to
the session timestamp."},"evidence":["D3:1"],"confidence":"high"}]}
```

The per-entity outputs are unioned and consolidated within the same session, so entity-centric extraction improves coverage without storing repeated copies of one memory.

Prompt K.3 Stage 3: session summary

Task. You are a session summarizer for a temporal personal context graph. Read one timestamped dialogue session and return exactly one compact, chronological, information-dense `session_summary`, using only the input.

Rules. Preserve important entities, events, preferences, relationships, plans, beliefs, motivations, and habits; omit greetings, closings, and temporary dialogue actions. The summary is at most 90 English words (or 140 Chinese characters). Attach exactly one time object: `session` when only the session time applies, `mixed` when the summary also contains event dates or ranges, and `unknown` only when no usable time exists. Normalize relative time against the session timestamp and always give a short temporal reason.

Runtime input.

```
Session timestamp: <ISO time>
<current_dialogue>
{{CURRENT_DIALOGUE}}
</current_dialogue>
```

Output. Return valid JSON only:

```
{
  "session_summary": {
    "summary": "",
    "time": {
      "value": [],
      "kind": "<session|mixed|unknown>",
      "confidence": "<high|medium|low>",
      "reason": ""
    },
    "evidence": []
  }
}
```

Example.

```
Input:
Session timestamp: 2023-03-27T13:10:00
D1:2 Andrew: I started a Financial Analyst job last week.
D1:7 Andrew: I love hiking; Fox Hollow is a great weekend trail.

Output:
{"session_summary":{
"summary":"Andrew said he started a Financial Analyst job last week
and recommended Fox Hollow as a weekend hiking trail.",
"time":{"value":[["2023-03-20","2023-03-26"],
["2023-03-27T13:10:00"]],"kind":"mixed","confidence":"high",
"reason":"Job start last week plus the current session."},
"evidence":["D1:2","D1:7"]}}
```

After the model returns `session_summary`, the program deterministically assembles that summary, the session timestamp, raw turn sequence, and committed fact identifiers into the corresponding episode cue; no additional generation call is used.

Prompt K.4 Intra-session consolidation

You are a fact-consolidation judge. All candidates were extracted entity-by-entity from the same source session.

Rules. Partition by core memory: merge only candidates with the same main subjects, relation/state/preference/plan/event, and semantic meaning. Keep related but distinct aspects separate; if unsure, keep them separate. Preserve every supported detail in a merged fact, choose the most representative topic and most specific time object, and do not perform cross-session timeline updating. Every input index must appear exactly once.

Input.

```
<candidate_facts>
{{CANDIDATE_FACTS}}
</candidate_facts>
```

Each item contains `index,fact,entities,topic,time,evidence`.

Prompt K.4 Intra-session consolidation (continued)

Output.

```
{"groups":[{
  "members": [],
  "fact": "",
  "topic": "",
  "time": {"value":[],"kind":"","confidence":"","reason":""}
}]}
```

Example.

```
Input:
0: Caroline attended an LGBTQ support group and found it powerful.
1: Caroline found her visit to the LGBTQ support group inspiring.
2: Caroline plans to research adoption agencies.

Output:
{"groups":[
 {"members":[0,1],
  "fact":"Caroline attended an LGBTQ support group and found it
  powerful and inspiring.","topic":"Caroline support group",
  "time":{"value":["2023-05-07"],"kind":"event",
  "confidence":"high","reason":"Same dated visit."}},
 {"members":[2],
  "fact":"Caroline plans to research adoption agencies.",
  "topic":"Caroline adoption plan",
  "time":{"value":["2023-06-09"],"kind":"plan_observed",
  "confidence":"high","reason":"Observed in the source session."}}
]}
```

The program, rather than the LLM, unions entity identifiers, evidence turns, and evidence-context windows. An invalid partition therefore cannot silently drop a source pointer.

K.2 Schema Formation Prompts

The four calls below separate coherence auditing, level-1 synthesis, cross-session grouping, and higher-level synthesis. The allowed basic categories are `update_constraint`, `privacy_constraint`, `profile_evidence`, and `episode_context`; strength is `hard`, `soft`, or `background`.

Prompt K.5 Level-1 coherence audit

You are a memory-consolidation judge. Audit whether embedding-nominated members describe one coherent event or episode.

Rules. Keep facts with the same core happening, plan, state, or interaction and main subjects. Drop a different event or an item that merely shares a person, place, or broad topic. A valid core has at least two members; `keep` and `drop` must cover every index exactly once.

Input.

```
<candidate_facts>
{{MEMBERS}}
</candidate_facts>
```

Each item contains `index,fact,topic,time,entities`.

Output.

```
{"same_event": <true|false>, "keep": [], "drop": []}
```

Example.

```
Input:
0: Caroline spoke about her transition at the school event.
1: Caroline encouraged students at the same school event.
2: Caroline plans to research adoption agencies.

Output:
{"same_event":true,"keep":[0,1],"drop":[2]}
```

Prompt K.6 Basic-gist synthesis

You are building a level-1 gist from one audited, coherent slice of a source session. Write the smallest future-useful operation concept, not a timeline or a copy of the facts.

Rules. A forget/revoke/update instruction states what a later answer must reject; sensitive data states what the answer should avoid repeating; quoted, fictional, or third-party material is not promoted to stable user-profile evidence. A durable preference states when it applies. Use only the accepted members and do not copy raw private identifiers.

Input.

```
<members>
{{MEMBERS}}
</members>
```

Each item contains `fact,topic,time,entities`.

Output.

```
{
 "topic": "",
 "category": "<update_constraint|privacy_constraint|
              profile_evidence|episode_context>",
 "slot": "",
 "constraint_strength": "<hard|soft|background>",
 "routing_cues": [],
 "text": "",
 "answer_effect": ""
}
```

Example.

```
Input:
- Earlier preference: early meetings are convenient.
- Latest instruction: do not schedule meetings before 10 a.m.

Output:
{"topic":"meeting-time update","category":"update_constraint",
"slot":"meeting_time","constraint_strength":"hard",
"routing_cues":["meeting time","early meeting","after 10 a.m."],
"text":"The latest instruction supersedes the earlier early-meeting
preference.","answer_effect":"Reject options that schedule the user
before 10 a.m."}
```

Prompt K.7 Higher-level grouping audit

You are a memory-consolidation judge. Given current forest roots, including orphan basic-gist roots, nominated by complete-linkage clustering, identify genuine cross-session themes.

Here, each orphan fact is serialized as the evidence-bearing payload of the special orphan basic-gist root defined in Appendix D.

Rules. Group at least two items only when they form one recurring storyline, project, relationship, multi-step plan, preference, or rule. Sharing a person or a broad word is insufficient. Each index appears in at most one group; every ungrouped index goes to `drop`.

Input.

```
<candidates>
{{CANDIDATES}}
</candidates>
```

Each item contains `index,kind,topic,text,time,entities`.

Output.

```
{"groups":[{"members":[],"reason":""}],"drop":[]}
```

Example.

```
Input:
0 gist: prefers quiet trails for low-impact weekends.
1 gist: prefers gentle park walks for restorative outdoor time.
2 fact: set a budget for a new laptop.

Output:
{"groups":[{"members":[0,1],
"reason":"Both encode the same recurring low-impact outdoor
preference."}],"drop":[2]}
```

Prompt K.8 Higher-level gist synthesis

You are maintaining a higher-level gist. Merge the proposed child gists only when they share one cross-session operation or schema with the same answer-time consequence; never write a story or timeline.

Input.

```
<children>
{{CHILDREN}}
</children>
```

Each child contains `topic,text,category,answer_effect`.

Output.

```
Accepted:
{"merge":true,"topic":"",
"category":"<higher_update_constraint|higher_privacy_constraint|
            higher_profile_evidence|higher_episode_context>",
"slot":"","constraint_strength":"<hard|soft|background>",
"routing_cues":[],"text":"","answer_effect":"",
"validity_boundary":""}

Declined:
{"merge":false,"reason":""}
```

Example.

```
Input children:
- quiet trails; prefer low-impact outdoor options
- gentle park walks; prefer restorative low-impact outdoor options

Output:
{"merge":true,"topic":"low-impact outdoor activity",
"category":"higher_profile_evidence","slot":"weekend_activity",
"constraint_strength":"soft",
"routing_cues":["quiet trail","gentle walk","outdoor weekend"],
"text":"The user repeatedly prefers quiet, low-impact outdoor
activities.","answer_effect":"Prefer gentle outdoor options.",
"validity_boundary":"Do not apply after an explicit newer mobility
restriction or activity update."}
```

Children sharing only the word *travel* must instead return `merge:false`.

This sequence separates candidate nomination, compatibility decisions, and gist synthesis. A declined group remains as separate roots and no fact text is altered.

Prompt K.9 Entity-profile prompt

Task. You are an entity-memory archivist. Build a compact memory record for one entity from only its time-sorted connected facts and evidence. This is a memory archive, not a dictionary or encyclopedic description.

Rules. The entity remains the center: state who did what with it and when. The `narrative_chain` is strictly chronological and contains one semicolon-separated node for every distinct date; facts sharing a date merge into one node, and undated facts form one final `[unknown]` node. The `memory_snapshot` is a 20–40 word person- and event-anchored paragraph, at most 50 words. Do not invent a generic meaning for an activity or concept, skip a dated fact, or repeat the chain verbatim.

Runtime input.

```
<entity_profile>
{{ENTITY_PROFILE}}
</entity_profile>
<facts>
{{FACTS_JSON}}
</facts>
```

Prompt K.9 Entity-profile prompt (continued)

Output. Return valid JSON only:

```
{
 "entity_id":"","name":"","type":"",
 "canonical_summary":{
   "narrative_chain":"",
   "memory_snapshot":"",
   "temporal_span":{"value":[],"confidence":"<high|medium|low>",
                    "reason":""}
 },
 "processed_fact_ids":[],
 "summary_confidence":"<high|medium|low>",
 "reason":""
}
```

Example.

```
Input facts:
[2023-05-08] Melanie used painting to express her feelings.
[2023-05-25] Melanie ran regularly for stress relief.

Output:
{"entity_id":"E7","name":"Melanie","type":"person",
"canonical_summary":{
"narrative_chain":"2023-05-08: used painting to express feelings;
2023-05-25: ran regularly for stress relief.",
"memory_snapshot":"Melanie uses painting for emotional expression and
running for stress relief across her May 2023 routines.",
"temporal_span":{"value":["2023-05-08","2023-05-25"],
"confidence":"high","reason":"Two dated source facts."}},
"processed_fact_ids":["F1","F2"],"summary_confidence":"high",
"reason":"All dated facts are represented."}
```

K.3 Answer-Generation Prompts

Prompt K.10 LoCoMo shared answer prompt

System. Answer the query using only the supplied retrieved memory. The memory may contain raw sessions, passages, summaries, or structured records. Select evidence by semantic relevance, speaker, event status, and local time. Interpret relative language only against an explicit local timestamp; distinguish completed events from plans, hypotheticals, and another person's experiences; prefer direct, specific, better-dated evidence; and never invent a precise date.

Runtime input.

```
<query>
{{QUERY}}
</query>
<retrieved_memory>
{{RETRIEVED_MEMORY}}
</retrieved_memory>
```

Output. Return valid JSON only:

```
{"used_evidence":[""],"answerable":<true|false>,"background":"",
"reasoning_summary":"","answer":""}
```

Use no more than five short evidence descriptions and keep the final answer at the granularity requested.

Example.

```
Query: How long had Caroline known these friends?
Memory: In the session dated 9 June 2023, Caroline said she had
known these friends for four years.

Output:
{"used_evidence":["Caroline said she had known the friends for four
years."],"answerable":true,"background":"The duration is explicit in
the dated session.","reasoning_summary":"The question asks for a
duration, not a calendar date.","answer":"About four years"}
```

All reported cross-method LoCoMo panels use this shared contract. The controlled adapter fills `RETRIEVED_MEMORY` with three raw sessions; the method-native Top-10 adapter fills it with each method's ranked native units; and the token-matched adapter fills the same field after exact 1,024-token matching. Within each comparison, only the serialized memory block changes.

Prompt K.11 PersonaMem-v2 answer prompt

Task. Choose the option that best answers the user's current query. Use sources in this order: (1) the current conversation as working context that has just happened, (2) retrieved long-term memory from completed earlier sessions, and (3) the benchmark-provided static persona. The current conversation has not yet been inserted into long-term memory.

Rules. Respect the latest explicit user preference or update when memories conflict; do not turn an unaccepted assistant suggestion into a user preference and do not invent private facts. When retrieved memory contains a structured `answer_effect`, apply it as a preference, avoidance, exclusion, or validity rule, followed by its supporting source-grounded facts. Retrieved facts and gists must satisfy the query's causal boundary. Choose exactly one of A, B, C, or D.

Runtime input.

```
<static_persona>{{STATIC_PERSONA}}</static_persona>
<current_conversation>{{CURRENT_CONVERSATION}}</current_conversation>
<retrieved_memory>{{RETRIEVED_MEMORY}}</retrieved_memory>
<query>{{QUERY}}</query>
<options>{{OPTIONS}}</options>
```

Output. Return valid JSON only:

```
{"choice":"<A|B|C|D>","reason":"brief grounded reason"}
```

Example.

```
Current conversation: asks for a low-impact weekend activity.
Prior memory: ongoing knee recovery; prefers quiet outdoor time.
Options:
A. Trail running
B. A gentle park walk
C. A crowded indoor concert
D. A high-intensity class

Output:
{"choice":"B","reason":"A gentle park walk fits the user's current
knee recovery and prior preference for quiet outdoor activity."}
```

Every method receives the same static persona, current dialogue, query, options, and multiple-choice output contract. The method-specific memory field contains its ranked representation; for CorMem, this is a compact schema constraint block followed by source-grounded facts.

K.4 Rewrite Ablation

Candidate targets are the Top-10 previously committed, causally available active facts whose embedding cosine is at least 0.92. The operation is applied before the next query's retrieval and defines the Rewrite comparison as an alternative update mechanism; it does not modify the append-only facts used by the CorMem schema variant.

Prompt K.12 Rewrite decision prompt

Task. You are a memory-update judge, not an extractor. Compare one newly extracted fact with causally available active historical fact blocks, then decide how the new fact updates the Rewrite memory. Use only the input; do not re-extract, discard a meaningful observation, resolve a contradiction by deleting one side, or inspect a future observation.

Decision rules. Default to `create_new`; related topic or shared entity alone is not a match. Use `exact_duplicate_confirm` only for the same entity, core relation, state/preference/plan/event, and meaning with no new detail, behavior, constraint, motivation, contradiction, or temporal development. Use `timeline_append_update` when the same core memory gains a detail, state, plan, preference, constraint, nuance, correction, or apparent contradiction. The latter preserves old and new statements as explicit `[time] text` segments. Merge aliases conservatively; keep named entities distinct from generic categories. Rename a semantic topic only when the new label is a narrower wording of the same memory aspect.

Runtime input.

```
<new_fact>
{{NEW_FACT_JSON}}
</new_fact>
<retrieved_fact_blocks>
{{RETRIEVED_FACT_BLOCKS_JSON}}
</retrieved_fact_blocks>
```

Prompt K.12 Rewrite decision prompt (continued)

Output. Return valid JSON only:

```
{
 "entity_updates":[{
   "action":"<merge_alias|keep_separate>",
   "new_entity":"","matched_existing_entity":"","canonical_name":"",
   "aliases_to_add":[],"confidence":"<high|medium|low>","reason":""
 }],
 "topic_update":{
   "action":"<keep_topic|rename_to_existing_topic|add_topic_alias>",
   "new_semantic_topic":"","canonical_semantic_topic":"",
   "aliases_to_add":[],"confidence":"<high|medium|low>","reason":""
 },
 "fact_decision":{
   "new_fact_id":"",
   "action":"<create_new|exact_duplicate_confirm|
              timeline_append_update>",
   "target_fact_ids":[],
   "final_active_fact":{
     "fact":"","semantic_topic":"","fact_type":"","relation_type":"",
     "entities":[],
     "time":{"value":[],"kind":"","confidence":"","reason":""},
     "evidence":[],"evidence_context":[],
     "confidence":"<high|medium|low>","status":"active"
   },
   "old_fact_metadata_updates":[{
     "fact_id":"","append_evidence":[],"append_evidence_context":[],
     "append_observation_time":{
       "value":[],"kind":"","confidence":"","reason":""},
     "append_source_session_id":"","reason":""
   }],
   "old_fact_status_updates":[{
     "fact_id":"","new_status":"<merged_into_new|archived>",
     "remove_from_active_retrieval":true
   }],
   "confidence":"<high|medium|low>","reason":""
 }
}
```

For an exact duplicate, `final_active_fact` is null and observation metadata are appended to the matched active fact.

Example.

```
Old:
{"fact_id":"F_old","fact":"Calvin visited Japan and enjoyed the
food and public transportation.","semantic_topic":"Calvin travel
experience","status":"active"}
New:
{"fact_id":"F_new","fact":"Calvin started learning Japanese because
he wants to travel more comfortably in Japan.",
"semantic_topic":"Calvin language learning","entities":["E_Calvin"],
"evidence":["D4:6"]}

Output:
{"entity_updates":[],
"topic_update":{"action":"keep_topic",
"new_semantic_topic":"Calvin language learning",
"canonical_semantic_topic":"Calvin language learning",
"aliases_to_add":[],"confidence":"high",
"reason":"Language learning is a distinct memory aspect."},
"fact_decision":{"new_fact_id":"F_new","action":"create_new",
"target_fact_ids":[],"final_active_fact":{
"fact":"Calvin started learning Japanese because he wants to travel
more comfortably in Japan.",
"semantic_topic":"Calvin language learning",
"fact_type":null,"relation_type":null,"entities":["E_Calvin"],
"time":{"value":["2023-06-09"],"kind":"motivation_observed",
"confidence":"high","reason":"Observed in the new source session."},
"evidence":["D4:6"],"evidence_context":[],
"confidence":"high","status":"active"},
"old_fact_metadata_updates":[],"old_fact_status_updates":[],
"confidence":"high",
"reason":"Related to travel, but not the same core memory."}}
```