DG-MEM: DUAL-GRAPH MEMORY RETRIEVAL FOR MULTI-USER AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term memory enables LLM agents to build on past interactions, but most existing systems are designed around a single user. Shared assistants in enterprise channels, project discussions, and organizational workflows must instead serve multiple users through a common conversational history. This introduces three challenges: distributed and heterogeneous evidence, coexisting and evolving user states, and asker-dependent relevance. Content-oriented retrieval and memory consolidation can miss cross-user connections or obscure the attribution needed to distinguish valid information simultaneously. We propose DG-MEM, a dual-graph framework that couples shared content organization with user-conditioned evidence selection. A content-centric graph connects evidence through semantic, conversational, and temporal relations, while a user-context graph preserves speaker-grounded distinctions through authorship and interaction structure. Aligned message anchors connect these complementary views. Asker-conditioned cross-graph retrieval accesses relevant evidence across participants, and query-conditioned refinement selects supporting messages according to user context and the question's specific requirements. The framework also supports conventional single-user memory, providing a common foundation for personal and shared histories. Memory construction and retrieval do not require generative-LLM calls and incur zero generative-LLM token cost before answer generation. Experiments on GroupMemBench and LoCoMo demonstrate state-of-the-art overall performance in multi-user and conventional single-user settings, respectively. Our code is available at https://anonymous.4open.science/r/DG-Mem.

1 INTRODUCTION

Long-term memory enables large language model (LLM) agents (DeepSeek-AI, 2026; Hurst et al., 2024; Yang et al., 2025) to retain information, build on past interactions, and support tasks that unfold over time. It is therefore a foundational capability for agents operating beyond isolated requests. Existing research has advanced along three major directions. Similarity-based systems (Lewis et al., 2020; Robertson & Zaragoza, 2009) retrieve historical messages or memory units through lexical or semantic matching. Graph-based systems organize entities, events, or memory records into relational structures to support relational and multi-hop retrieval (Gutierrez et al., 2024; Yang et al., 2026a). Agentic memory systems use LLMs or dedicated controllers to extract, summarize, update and reorganize experience (Xu et al., 2025; Yan et al., 2025; Packer et al., 2024). These approaches provide increasingly sophisticated ways to organize and access historical information. However, much of this progress has been centered on dyadic interactions, where one agent serves one user. Shared collaborative environments extend this setting to histories generated and accessed by multiple participants, introducing new requirements for how memories are represented and retrieved.

In enterprise channels, project discussions, and organizational workflows, a shared agent must serve multiple users through a common conversational history. As illustrated in Figure 1, Daniel's Monday deadline and James's Friday deadline concern the same project but answer different users' questions. Such settings introduce three challenges. ❶ **Distributed and heterogeneous evidence.** Supporting information spans channels and reply threads, while role-specific vocabulary creates lexical gaps between questions and relevant history. Similarity-based retrieval can consequently miss evidence whose relevance emerges through conversational connections. ❷ **Coexisting and evolving user states.** Shared decisions and individual commitments can remain simultaneously valid, with later

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.

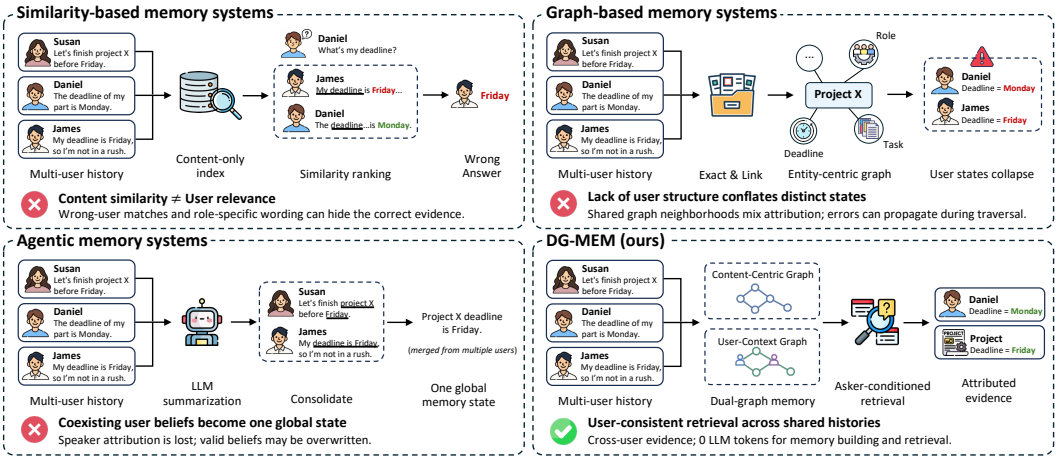

Figure 1: Existing similarity-based, graph-based, and agentic memory systems struggle in multi-user settings. DG-MEM combines complementary content and user-context graphs for asker-conditioned retrieval, preserving attribution across shared conversational histories.

updates affecting only particular users or tasks. Entity-centric aggregation and LLM-based consolidation can obscure authorship and contextual scope, merging distinctions that must remain accessible. ❸ **Asker-dependent relevance.** The same question can require different answers depending on who asks it. Topically similar messages or nearby graph nodes may concern another user's state, making content relevance alone insufficient for evidence selection. Retrieving the appropriate evidence requires combining shared conversational context with the asker's identity and the question's requirements. Together, these challenges call for memory systems that preserve shared associations while grounding evidence in user and interaction context (Yang et al., 2026b). Repeated LLM-based extraction, summarization, and consolidation further increase token costs as shared histories grow.

To address these challenges, we propose DG-MEM, a dual-graph framework that couples shared content organization with user-conditioned evidence selection. To connect evidence distributed across the history, a Content-Centric Memory Graph organizes messages through semantic, reply, and temporal relations. A complementary User-Context Graph preserves distinctions among users and contexts by organizing the same messages around authors, roles, replies, and conversational scopes, keeping shared decisions and individual commitments separately attributable. Aligned message anchors connect the two views, allowing each piece of evidence to retain both its content associations and participant context. Asker-Conditioned Cross-Graph Retrieval transfers content-relevant candidates into the user-context view and guides propagation using the asker and query-matched contexts. This enables access to relevant evidence from other participants while grounding retrieval in the current user's context. Query-Conditioned Candidate Refinement then combines content and interaction relevance with the question's requested fields and temporal or state cues, prioritizing candidates that support the specific answer requirements. Together, temporal associations and query-conditioned refinement support evidence selection as decisions and commitments evolve. The content view also supports conventional single-user memory, providing a common foundation for personal and shared histories. Memory construction and retrieval require no generative-LLM calls and incur zero LLM token cost before answer generation. Our contributions are threefold:

- We introduce a dual-graph memory representation that separates content organization from user-context structure and connects the two views of the same underlying conversational history through aligned message anchors. This design preserves shared conversational associations and speaker-grounded interactions within a common framework for personal and multi-user memory.

- We develop DG-MEM, which transfers content-relevant candidates across graph views and combines asker-conditioned propagation with query-conditioned refinement to select supporting evidence for the current user's question. Memory construction and retrieval require no generative-LLM calls, incurring zero generative-LLM token cost before answer generation.

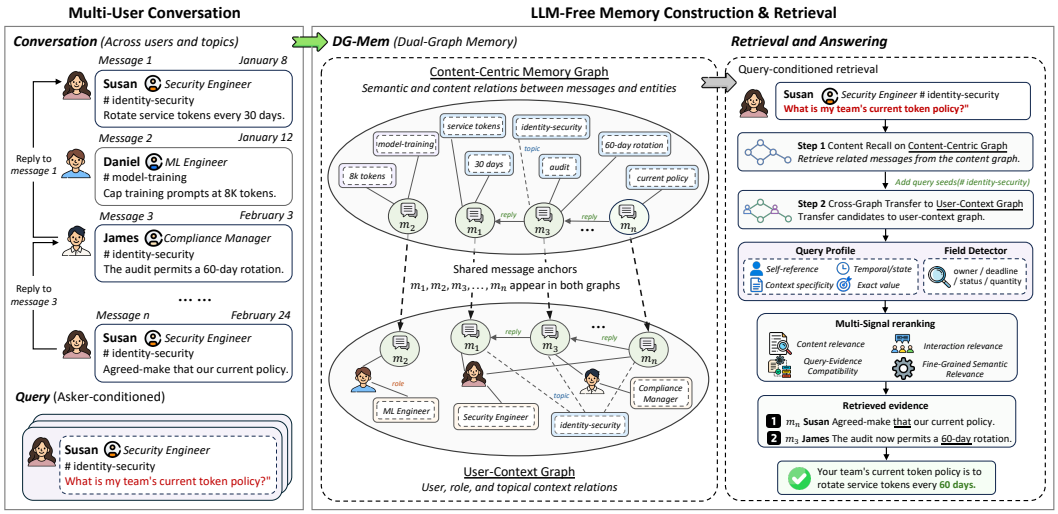

Figure 2: Overview of DG-MEM. Content-relevant candidates are transferred between complementary graphs through shared message anchors, then refined using asker context and query-conditioned multi-signal reranking. The example links replies across users to recover the updated token policy in the correct context. Memory construction and retrieval require no generative-LLM calls.

- We conduct comprehensive experiments on GroupMemBench and LoCoMo, demonstrating state-of-the-art performance in multi-user memory and validating the framework's effectiveness in single-user settings, while achieving substantial reductions in memory construction time.

2 PRELIMINARIES

We consider a general long-term conversational memory setting with a set of users $\mathcal{U} = \{u_1, \ldots, u_K\}$ and a memory-augmented agent u_a. Following the multi-party formulation of GroupMemBench (Yang et al., 2026b), the interaction history observed up to time T is

$$\mathcal{C}_{1:T} = (m_1, \ldots, m_T), \qquad m_i = (a_i, t_i, x_i, c_i), \tag{1}$$

where $a_i \in \mathcal{U} \cup \{u_a\}$ is the author, t_i is the timestamp, x_i is the message content, and c_i denotes the available conversational metadata, such as the channel, topic, author role, and reply target. This definition permits both user–agent and user–user interactions without assuming a particular memory representation. A memory system constructs a state $\mathcal{M}_T$ from the observed history:

$$\mathcal{M}_T = F_{\text{mem}}\left(\mathcal{C}_{1:T}\right). \tag{2}$$

At query time, an asking user, or *asker*, $u_q \in \mathcal{U}$ issues a question q. The agent produces an answer

$$\hat{y} = F_{\text{ans}}\left(q, u_q, \mathcal{M}_T\right). \tag{3}$$

In the conventional dyadic setting, the asker is fixed and its identity is often omitted. In the multi-user setting, u_q is a variable input: the interpretation of a query and the relevant portion of memory may depend on the asker's identity and interaction history. The formulation itself imposes no requirement on how $\mathcal{M}_T$ is constructed; it may be an unstructured/structured store, or a set of summaries.

3 METHOD

3.1 OVERVIEW

We propose DG-MEM, a dual-graph framework that represents a conversational history through complementary content and interaction views. The Content-Centric Memory Graph $\mathcal{G}^M$ captures semantic and conversational associations among messages and produces an initial content-relevant candidate set. The User-Context Graph $\mathcal{G}^U$ organizes the same messages around users, roles, replies,

and conversational contexts. Their message anchors are aligned by message identity, allowing evidence retrieved in one view to be located directly in the other. Given (q, u_q), DG-MEM first retrieves from $\mathcal{G}^M$ and transfers the resulting candidate ranks as seed mass to $\mathcal{G}^U$. The asker and query-matched contexts then guide structural propagation over the second graph. A query-conditioned fusion stage combines content, interaction, and fine-grained relevance to produce the final evidence ranking. Together, the two graph views instantiate $\mathcal{M}_T$. Memory construction and retrieval use local encoders and deterministic graph propagation, incurring zero generative-LLM token cost.

3.2 CONTENT-CENTRIC MEMORY GRAPH

The first retrieval stage seeks evidence related to the content of the query across the complete conversational history. Such evidence exhibits semantic continuity when different messages refer to the same objects, as well as conversational continuity through replies and temporal progression. The Memory Graph captures complementary structures by treating messages as evidence units, entities as semantic junctions, and conversational relations as paths between nearby pieces of evidence.

$$\mathcal{G}^M = \left(V_m \cup V_e,\, E^{\text{ent}} \cup E^{\text{reply}} \cup E^{\text{temp}} \cup E^{\text{phase}}\right), \tag{4}$$

where $V_m = \{v_1, \ldots, v_T\}$ contains one anchor v_i for each message m_i, and V_e contains normalized entity surfaces extracted from message content. Messages that repeat the same normalized entity surface are connected through a common entity node. The corresponding message–entity edge is undirected and weighted by the normalized occurrence count within the message. The remaining relations preserve local conversational structure among individual messages recorded in the history. An undirected edge in E^{reply} connects a message with its reply parent. E^{temp} contains directed edges between chronologically consecutive messages within the same channel, with edge strength decreasing as their temporal distance grows. When adjacent messages share a non-empty interaction phase, E^{phase} adds an undirected connection between them. These relations provide complementary paths across repeated semantic references and the local progression of the conversation.

Given a query q, we use personalized PageRank (PPR) to propagate relevance through the Memory Graph for evidence retrieval. Let $\mathbf{p}^M$ denote the normalized query-dependent seed distribution, P^M the row-normalized transition matrix with zero dangling rows, and $\mathbf{d}^M$ the dangling-node indicator. With damping factor $\alpha_M \in (0, 1)$, the node relevance distribution is updated as:

$$\boldsymbol{\pi}^{M,(t+1)} = (1 - \alpha_M)\mathbf{p}^M + \alpha_M \left[(P^M)^\top \boldsymbol{\pi}^{M,(t)} + \left((\mathbf{d}^M)^\top \boldsymbol{\pi}^{M,(t)}\right) \mathbf{p}^M\right], \tag{5}$$

where t indexes iterations and $\boldsymbol{\pi}^{M,(0)} = \mathbf{p}^M$. The final component π_i^M gives the propagation-based relevance of message anchor v_i. Next, we fuse graph, semantic, and lexical relevance over the union of top-ranked dense, PPR, and BM25 candidates. Let $\widehat{\mathbf{s}}_i^M = [\widehat{\pi}_i^M, \widehat{s}_i^{\text{den}}, \widehat{s}_i^{\text{bm}}]^\top$ collect the PPR, cosine similarity, and BM25 scores, each normalized within this candidate set. Constant-score components are mapped to zero. Using fixed convex fusion weights $\boldsymbol{\omega}^M$, we compute:

$$S_i^M(q) = (\boldsymbol{\omega}^M)^\top \widehat{\mathbf{s}}_i^M. \tag{6}$$

The fusion coefficients are specified in the Appendix A.5. The highest-ranked messages form the content proposal $\mathcal{R}^M(q)$ for subsequent asker-conditioned retrieval through the interaction view.

3.3 USER-CONTEXT GRAPH

Content relevance identifies what parts of the history may answer a question, but semantically similar messages can describe different users or different interaction contexts within the same history. The User-Context Graph resolves this ambiguity by embedding the aligned message anchors in a relational structure centered on participation, discourse, and contextual affiliation.

$$\mathcal{G}^U = \left(V_m \cup V_u \cup V_r \cup V_s,\, E^{\text{part}} \cup E^{\text{disc}} \cup E^{\text{aff}}\right), \tag{7}$$

where V_u and V_r represent observed users and roles, and V_s represents conversational scopes. Each scope summarizes the available contextual fields associated with a message and is instantiated in our implementation by its channel, phase, and topic. Because both graphs use the message index V_m to represent the same set of messages, the score associated with v_i in the Memory Graph can be transferred to the corresponding anchor in the User-Context Graph. Participant relations E^{part}

connect each message with its author and with users explicitly mentioned in its content. Discourse relations E^{disc} connect messages with their reply parents and conversational scopes. Affiliation relations E^{aff} associate users with observed roles and with the scopes in which they participate. User–scope weights grow sublinearly with repeated participation, preserving recurring interactions while controlling the influence of highly active users. These relations are undirected and collectively place each message within its participant and interaction neighborhood.

The graph is accompanied by a context-aware BM25 index that combines message content with available author, role, and scope fields. This lexical view captures direct contextual matches, while graph propagation captures connections through users, replies, roles, and shared scopes.

3.4 Asker-Conditioned Cross-Graph Retrieval

Aligned message anchors connect the two graph views, allowing the content proposal $\mathcal{R}^M(q)$ to initialize retrieval over $\mathcal{G}^U$. Content candidates are mapped to their corresponding message anchors and assigned rank-decayed seed mass. Messages retrieved by the context-aware BM25 index contribute additional rank-decayed mass, combining content-based discovery with direct contextual matching.

To condition propagation on the interaction represented by (q, u_q), we additionally seed the asker, users explicitly identified in the query, and query-matched role and scope nodes. For self-referential queries, an asker-specific author token is added to the lexical query, and asker personalization is strengthened. These weighted contributions are accumulated into a nonnegative, query-dependent seed vector $\mathbf{z}^U(q, u_q)$ over all nodes of $\mathcal{G}^U$. The accumulated mass is jointly normalized to obtain the personalization distribution used to guide subsequent propagation through the graph:

$$\mathbf{p}^U = \frac{\mathbf{z}^U(q, u_q)}{\|\mathbf{z}^U(q, u_q)\|_1}. \tag{8}$$

Joint normalization preserves the relative mass contributed by different sources in the combined seed distribution. If no seeds are activated, we use a uniform distribution over message nodes. The weighting and activation rules are provided in the Appendix A.5. Applying the PPR update in Eq. 5 to $\mathcal{G}^U$, using its transition structure and $\mathbf{p}^U$ for initialization and personalization, yields the node relevance distribution $\boldsymbol{\pi}^U$. Content seeds identify relevant message neighborhoods, while user and context seeds guide propagation through authorship, replies, mentions, roles, and shared scopes. These paths allow evidence from other participants to become accessible under the asker's context. The resulting candidate set is the union of $\mathcal{R}^M(q)$, the context-aware BM25 results, and the highest-ranked message anchors under $\boldsymbol{\pi}^U$. These candidates, together with their available content and interaction relevance signals, are passed to query-conditioned refinement.

3.5 Query-Conditioned Candidate Refinement

Refinement prioritizes structurally retrieved candidates according to the question's answer requirements. Lightweight lexical rules identify self-reference and contextual, temporal, state-oriented, and exact-answer cues, together with the requested answer field. Candidates are evaluated through four evidence families: content relevance from the Memory Graph; interaction relevance from context-aware BM25, User-Context Graph propagation, and asker compatibility through authorship, replies, and mentions; query–evidence compatibility through contextual overlap, evidential-status cues, and field matching; and fine-grained semantic relevance from a local sentence-level cross-encoder. Reranking retains the highest-scoring sentence, restricting evaluation to sentences with the strongest field match when field-matching evidence is available.

These signals form a candidate feature vector $\boldsymbol{\phi}_i(q, u_q)$. Memory Graph, BM25, PPR, and cross-encoder scores are min–max normalized over the candidate set, while the remaining components use bounded rule-based scores. A query-conditioned fusion produces the refinement score:

$$S_i^{\text{ref}}(q, u_q) = \boldsymbol{\omega}^{\text{ref}}(q, u_q)^\top \boldsymbol{\phi}_i(q, u_q), \tag{9}$$

where $\boldsymbol{\omega}^{\text{ref}}(q, u_q)$ is a convex weighting vector obtained through predefined query-dependent adjustments to shared base weights. Self-referential queries favor direct asker compatibility relative to broad graph proximity. Contextual queries activate scope matching, while temporal, state-oriented, and exact-answer queries activate evidential-status signals. Field compatibility contributes when a

Table 1: Accuracy (%) on GroupMemBench across six query categories. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	36.26	23.36	21.70	46.94	32.72	75.23	38.04
BM25	40.11 (+3.85)	25.23 (+1.87)	14.15 (−7.55)	40.82 (−6.12)	54.94 (+22.22)	77.98 (+2.75)	43.22 (+5.18)
GraphRAG	12.09 (−24.17)	14.02 (−9.34)	19.81 (−1.89)	14.29 (−32.65)	5.56 (−27.16)	66.97 (−8.26)	20.56 (−17.48)
HippoRAG	39.56 (+3.30)	27.10 (+3.74)	30.19 (+8.49)	42.86 (−4.08)	29.63 (−3.09)	75.23 (0.00)	39.72 (+1.68)
Agent Memory Systems							
Mem0	21.98 (−14.28)	4.67 (−18.69)	11.32 (−10.38)	20.41 (−26.53)	16.67 (−16.05)	82.57 (+7.34)	25.73 (−12.31)
MemGPT	22.53 (−13.73)	17.76 (−5.60)	20.75 (−0.95)	28.57 (−18.37)	12.42 (−20.30)	77.98 (+2.75)	28.15 (−9.89)
A-Mem	35.16 (−1.10)	22.43 (−0.93)	26.42 (+4.72)	46.94 (0.00)	23.46 (−9.26)	67.89 (−7.34)	35.10 (−2.94)
LightMem	4.95 (−31.31)	0.00 (−23.36)	2.83 (−18.87)	6.12 (−40.82)	3.09 (−29.63)	75.54 (+0.31)	16.78 (−21.26)
SimpleMem	25.82 (−10.44)	8.41 (−14.95)	12.26 (−9.44)	20.41 (−26.53)	27.78 (−4.94)	64.75 (−10.48)	28.72 (−9.32)
Hindsight	42.31 (+6.05)	17.76 (−5.60)	37.74 (+16.04)	40.82 (−6.12)	54.94 (+22.22)	77.06 (+1.83)	46.01 (+7.97)
DG-MEM	52.20 (+15.94)	43.00 (+19.64)	32.08 (+10.38)	57.14 (+10.20)	53.09 (+20.37)	88.49 (+13.26)	55.30 (+17.26)

specific answer field is identified. Finally, greedy selection balances the refinement score against textual overlap and repeated authorship among selected messages. The selected evidence and its attribution metadata are supplied to the reader. The details are provided in the Appendix A.5.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTING

Datasets. We evaluate DG-MEM on two complementary conversational-memory benchmarks. 1) **GroupMemBench** (Yang et al., 2026b) targets long-term memory in multi-user conversations. It comprises 120K synthetic messages from enterprise communication channels spanning Finance, Technology, Healthcare, and Manufacturing, and evaluates six capabilities: multi-hop reasoning, knowledge update, term ambiguity, user-implicit reasoning, temporal reasoning, and abstention. Each query is issued by a specific user, enabling systematic evaluation of group dynamics, speaker-grounded memory, and audience-adapted language. 2) **LoCoMo** (Maharana et al., 2024) is a widely adopted benchmark for long-term memory over extended, multi-session dyadic conversations. Following prior work (Yan et al., 2025), we evaluate its single-hop, multi-hop, temporal-reasoning, and open-domain questions. Because each LoCoMo conversation involves a fixed pair of speakers rather than a group with a varying asker, we use it to assess whether DG-MEM also generalizes to the conventional dyadic setting used for evaluating single-user agent memory.

Baselines. We compare DG-MEM against multiple representative baselines, organized into two groups. 1) **RAG-based methods** comprise BM25, dense retrieval using text-embedding-3-large, GraphRAG (Edge et al., 2025), and HippoRAG (Gutierrez et al., 2024). BM25 and dense retrieval rank textual memory units according to lexical and semantic relevance, respectively, while GraphRAG and HippoRAG use graph-structured indices to retrieve related evidence. 2) **Agent memory systems** comprise Mem0 (Chhikara et al., 2025), MemGPT (Packer et al., 2024), A-Mem (Xu et al., 2025), Hindsight (Latimer et al., 2025), LightMem (Fang et al., 2026), GAM (Yan et al., 2025), MemoryOS (Kang et al., 2025), CompassMem (Hu et al., 2026) and SimpleMem (Liu et al., 2026). These methods maintain persistent memory states over interaction histories and selectively access stored information during inference to support contextually grounded answer generation. Additional descriptions and implementation details are provided in the Appendix A.7.

Implementation Details. For GroupMemBench, we follow the original protocol (Yang et al., 2026b). Baselines requiring LLM-based memory processing use GPT-4o-mini, GPT-5 is used for answer generation and evaluation, and embedding-based baselines use text-embedding-3-large. DG-MEM uses spaCy for NER and `all-MiniLM-L6-v2` for message and entity embeddings. For LoCoMo, following prior work (Yan et al., 2025), we use GPT-4o-mini and Qwen2.5-14B as the answering LLMs, BGE-M3 for embeddings, and at most five retrieved memory units per query.

Table 2: Performance on LoCoMo across four question types using F1 and BLEU-1. Parenthesized values are absolute score differences from LONG-LLM under the same base LLM, computed from the displayed scores. Green/red indicates improvement/degradation.

Method	Single Hop		Multi Hop		Temporal		Open Domain		Average	
	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1	F1	BLEU-1
Base LLM: GPT-4o-mini										
LONG-LLM	46.68	37.54	29.23	22.76	25.97	19.42	16.87	13.70	37.31	29.57
RAG	52.45 (+5.77)	47.94 (+10.40)	27.50 (−1.73)	20.13 (−2.63)	46.07 (+20.10)	40.35 (+20.93)	23.23 (+6.36)	17.94 (+4.24)	44.73 (+7.42)	39.40 (+9.83)
HippoRAG	54.84 (+8.16)	48.84 (+11.30)	33.59 (+4.36)	25.46 (+2.70)	48.17 (+22.20)	39.32 (+19.90)	28.59 (+11.72)	23.89 (+10.19)	47.92 (+10.61)	41.02 (+11.45)
A-Mem	44.65 (−2.03)	37.06 (−0.48)	27.02 (−2.21)	20.09 (−2.67)	45.85 (+19.88)	36.67 (+17.25)	12.14 (−4.73)	12.00 (−1.70)	39.65 (+2.34)	32.31 (+2.74)
Mem0	47.65 (+0.97)	38.72 (+1.18)	38.72 (+9.49)	27.13 (+4.37)	48.93 (+22.96)	40.51 (+21.09)	28.64 (+11.77)	21.58 (+7.88)	45.10 (+7.79)	35.90 (+6.33)
MemoryOS	48.62 (+1.94)	42.99 (+5.45)	35.27 (+6.04)	25.22 (+2.46)	41.15 (+15.18)	30.76 (+11.34)	20.02 (+3.15)	16.52 (+2.82)	42.84 (+5.53)	35.54 (+5.97)
LightMem	41.79 (−4.89)	37.83 (+0.29)	29.78 (+0.55)	24.80 (+2.04)	43.71 (+17.74)	39.72 (+20.30)	16.89 (+0.02)	13.92 (+0.22)	38.44 (+1.13)	34.36 (+4.79)
SimpleMem	53.48 (+6.80)	47.59 (+10.05)	36.93 (+7.70)	28.80 (+6.04)	51.30 (+25.33)	45.29 (+25.87)	21.78 (+4.91)	15.67 (+1.97)	48.02 (+10.71)	41.68 (+12.11)
CompassMem	57.36 (+10.68)	49.79 (+12.25)	38.84 (+9.61)	27.98 (+5.22)	57.96 (+31.99)	50.51 (+31.09)	26.61 (+9.74)	20.01 (+6.31)	52.18 (+14.87)	44.09 (+14.52)
GAM	57.75 (+11.07)	52.10 (+14.56)	42.29 (+13.06)	34.44 (+11.68)	59.45 (+33.48)	53.11 (+33.69)	33.30 (+16.43)	26.97 (+13.27)	53.75 (+16.44)	47.51 (+17.94)
DG-MEM	69.55 (+22.87)	63.32 (+25.78)	44.24 (+15.01)	33.79 (+11.03)	60.61 (+34.64)	54.11 (+34.69)	33.15 (+16.28)	27.39 (+13.69)	60.79 (+23.48)	53.75 (+24.18)
Base LLM: Qwen2.5-14B										
LONG-LLM	46.05	39.56	32.08	24.46	30.51	24.45	14.89	11.41	38.31	31.90
RAG	47.87 (+1.82)	42.79 (+3.23)	26.38 (−5.70)	19.54 (−4.92)	30.78 (+0.27)	25.97 (+1.52)	14.16 (−0.73)	10.52 (−0.89)	38.27 (−0.04)	33.01 (+1.11)
HippoRAG	42.45 (−3.60)	37.14 (−2.42)	27.57 (−4.51)	20.62 (−3.84)	30.66 (+0.15)	26.33 (+1.88)	19.74 (+4.85)	15.81 (+4.40)	35.85 (−2.46)	30.53 (−1.37)
A-Mem	33.75 (−12.30)	30.04 (−9.52)	22.09 (−9.99)	15.28 (−9.18)	27.19 (−3.32)	22.05 (−2.40)	13.49 (−1.40)	10.74 (−0.67)	28.98 (−9.33)	24.47 (−7.43)
Mem0	42.58 (−3.47)	35.15 (−4.41)	31.73 (−0.35)	24.82 (+0.36)	28.96 (−1.55)	26.24 (+1.79)	15.03 (+0.14)	11.28 (−0.13)	36.04 (−2.27)	29.91 (−1.99)
MemoryOS	46.33 (+0.28)	41.62 (+2.06)	38.19 (+6.11)	29.26 (+4.80)	32.24 (+1.73)	27.86 (+3.41)	20.27 (+5.38)	15.94 (+4.53)	40.28 (+1.97)	34.89 (+2.99)
LightMem	34.92 (−11.13)	31.22 (−8.34)	25.45 (−6.63)	19.61 (−4.85)	32.03 (+1.52)	27.70 (+3.25)	15.81 (+0.92)	11.81 (+0.40)	31.39 (−6.92)	27.15 (−4.75)
SimpleMem	51.11 (+5.06)	45.47 (+5.91)	34.04 (+1.96)	25.65 (+1.19)	48.46 (+17.95)	36.07 (+11.62)	23.41 (+8.52)	21.45 (+10.04)	45.71 (+7.40)	38.39 (+6.49)
CompassMem	61.02 (+14.97)	55.93 (+16.37)	42.32 (+10.24)	32.66 (+8.20)	47.18 (+16.67)	39.69 (+15.24)	25.88 (+10.99)	22.01 (+10.60)	52.52 (+14.21)	46.17 (+14.27)
GAM	58.93 (+12.88)	53.74 (+14.18)	42.96 (+10.88)	34.48 (+10.02)	51.52 (+21.01)	44.43 (+19.98)	30.63 (+15.74)	26.04 (+14.63)	52.70 (+14.39)	46.55 (+14.65)
DG-MEM	70.72 (+24.67)	64.96 (+25.40)	44.23 (+12.15)	33.64 (+9.18)	58.05 (+27.54)	51.78 (+27.33)	39.35 (+24.46)	35.25 (+23.84)	61.27 (+22.96)	54.63 (+22.73)

4.2 MAIN RESULTS

Multi-user scenario. Table 1 shows that DG-MEM achieves the best overall performance on GroupMemBench, with a question-count-weighted accuracy of 55.30%. This exceeds Hindsight, the strongest agent memory baseline, by 9.29 percentage points, corresponding to a 20.2% relative improvement, and surpasses BM25, the strongest RAG-based baseline, by 12.08 points. Beyond this substantial aggregate advantage, DG-MEM achieves the highest accuracy in four of the six query categories. The largest improvement is observed in Knowledge Update, where DG-MEM reaches 43.00%, outperforming the strongest baseline by 15.90 points. It also achieves 57.14% on User-Implicit Reasoning and 52.20% on Multi-Hop Reasoning, improving over the respective best baselines by 10.20 and 9.89 points. These gains directly address central demands of multi-user memory: recovering evolving group information, resolving asker-dependent references, and connecting evidence distributed across conversations. The improvement across these complementary tasks is consistent with the benefits of combining content relevance with user-context structure. DG-MEM further achieves 88.49% on Abstention, exceeding Mem0 by 5.92 points and demonstrating stronger performance when the requested information is unavailable. All these results are obtained with zero generative-LLM token consumption during memory construction and retrieval.

Single-user scenario. Table 2 demonstrates that DG-MEM also achieves the best average F1 and BLEU-1 on LoCoMo with both backbone models. With GPT-4o-mini, DG-MEM obtains 60.79 F1 and 53.75 BLEU-1, exceeding GAM, the strongest baseline on both average metrics, by 7.04 and 6.24 points, respectively. With Qwen2.5-14B, these scores reach 61.27 and 54.63, widening the respective gains over GAM to 8.57 and 8.08 points. The advantage extends across question types: DG-MEM achieves the highest F1 in seven of the eight task–backbone settings, including all four categories with Qwen2.5-14B. Single-Hop F1 improves over the strongest baseline by 11.80 points with GPT-4o-mini and 9.70 points with Qwen2.5-14B. These results establish the framework's effectiveness in conventional single-user conversations across different answering models, supporting its applicability to both personal and shared conversational memory.

4.3 ROBUSTNESS TO DIFFERENT LLM

We evaluate robustness to the choice of base LLM on GroupMemBench. As shown in Figure 3, DG-MEM achieves the highest accuracy across all four models, outperforming the strongest baseline by 3.36, 4.03, 1.74, and 2.95 percentage points with Qwen3-8B, GPT-4o-mini, Qwen3-32B, and DeepSeek-V4-Flash, respectively. Although the strongest baseline shifts from BM25 on Qwen3-8B

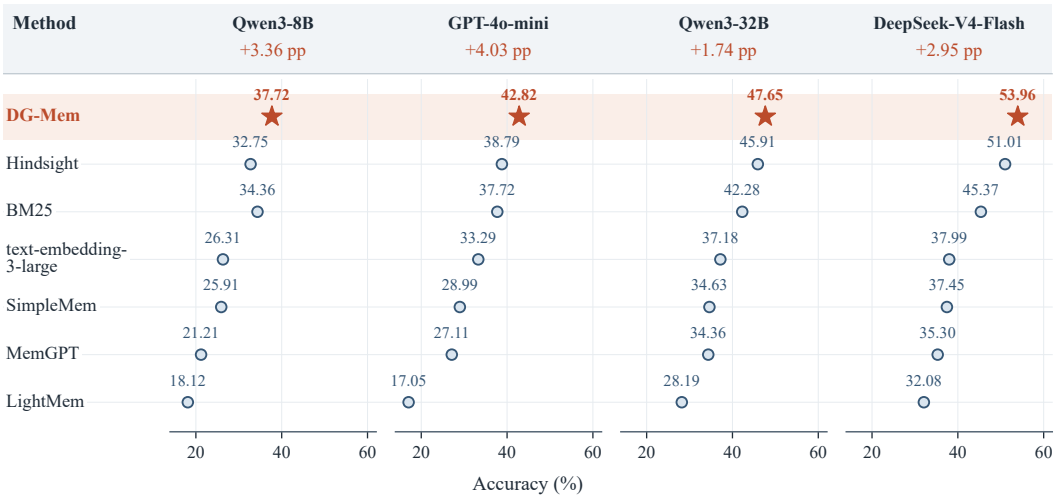

Figure 3: Average accuracy (%) on GroupMemBench across four base LLMs.

to Hindsight on the remaining models, DG-MEM consistently ranks first. Its advantage holds across both Qwen model sizes and extends to the GPT and DeepSeek families, supporting the effectiveness of the framework across different answering models. Within the Qwen family, DG-MEM improves from 37.72% to 47.65% as model size increases from 8B to 32B. With Qwen3-8B, it also matches BM25 with GPT-4o-mini at 37.72%, highlighting the importance of memory design alongside the answering model for question answering over extended interaction histories. Since memory construction and retrieval require no generative-LLM calls, DG-MEM provides a memory layer that can support different downstream LLMs while maintaining consistent performance gains.

4.4 ABLATION STUDY

We evaluate the contribution of each major component by removing it individually from DG-MEM. Figure 4 shows that the full framework outperforms all ablated variants. Removing the User-Context Graph causes the largest accuracy drop of 15.0 percentage points, highlighting the importance of modeling users and their interaction contexts in shared conversational histories. Removing the Content-Centric Memory Graph reduces accuracy by 9.7 points, demonstrating the contribution of semantic and conversational associations to evidence retrieval. These results support the complementary roles of content organization and user-context modeling for retrieving evidence from shared conversational histories. Disabling Asker-Conditioned Cross-Graph Retrieval also reduces accuracy, providing evidence for the benefit of conditioning cross-graph retrieval on the asker. Finally, removing Query-Conditioned Candidate Refinement yields a 5.2-point drop, demonstrating the value of integrating complementary relevance signals according to the query's evidence requirements. The strongest performance therefore comes from combining the two graph views with conditioned retrieval and candidate refinement.

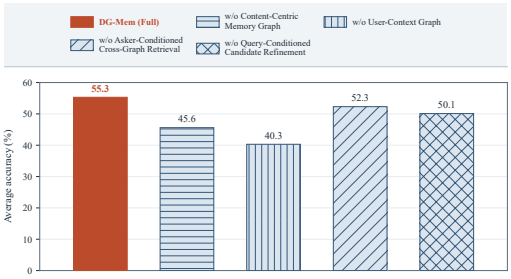

Figure 4: Ablation studies on GroupMemBench.

4.5 EFFICIENCY ANALYSIS

Table 3 compares the memory construction costs of DG-MEM with Hindsight, LightMem, and SimpleMem. DG-MEM requires zero generative-LLM calls and tokens, constructing memory through embedding computation and deterministic graph operations. Construction takes only 0.76 hours, reducing wall-clock time by 96.96% relative to SimpleMem, the fastest baseline in this compari-

Table 3: Memory construction efficiency compared with the most efficient and effective baselines on GroupMemBench. All methods run in the same environment with the same concurrency level.

Method	LLM Calls	LLM Tokens (M)	Wall-clock (h)	Memory Store (MB)	Accuracy
Hindsight	6,712	116.1	176.4	1,554	46.01
LightMem	27,815	20.9	26.0	705	16.78
SimpleMem	4,864	35.0	25.0	724	28.72
DG-MEM	**0**	**0**	**0.76**	**345**	**55.30**
Relative change	**↓100.0%**	**↓100.0%**	**↓96.96%**	**↓51.06%**	**↑20.19%**

son. The resulting memory store occupies 345 MB, a 51.06% reduction relative to LightMem, the most storage-efficient baseline. These efficiency gains accompany superior downstream accuracy: DG-MEM achieves 55.30%, exceeding Hindsight, the strongest baseline, by 9.29 percentage points (20.19% relative improvement). Thus, DG-MEM improves memory construction efficiency and storage compactness while delivering more accurate answers in multi-user conversations.

5 Related Work

Similarity-Based Memory Retrieval. Similarity-based systems retrieve past messages or memory units using lexical matching, as in BM25 (Robertson & Zaragoza, 2009), or dense semantic similarity (Lewis et al., 2020). Their scalability and simple construction make them common baselines for long-term memory in conversational agents with increasingly long interaction histories.

Graph-Structured Memory. Graph-based methods encode relations among memories to facilitate access to related information. GraphRAG (Edge et al., 2025) organizes extracted entities and relations into communities, while HippoRAG (Gutierrez et al., 2024) propagates relevance over an entity graph. Zep (Rasmussen et al., 2025) tracks temporally evolving relations, and Compass-Mem (Hu et al., 2026) connects events through explicit logical dependencies. These structures support relational and multi-hop retrieval beyond independent similarity scores.

Agentic Memory Management. Agentic systems use LLMs or dedicated controllers to construct and access memory. MemGPT (Packer et al., 2024) and MemoryOS (Kang et al., 2025) manage information across memory tiers. Mem0 (Chhikara et al., 2025) and A-Mem (Xu et al., 2025) extract and maintain evolving memories. GAM (Yan et al., 2025), Hindsight (Latimer et al., 2025), LightMem (Fang et al., 2026), and SimpleMem (Liu et al., 2026) further introduce query-time reconstruction, reflection, offline consolidation, and intent-aware retrieval, respectively. Such flexibility often introduces generative processing into the memory lifecycle.

Across these directions, existing systems are predominantly designed for a single user or fixed conversational dyad. User identifiers commonly partition histories into independent stores, leaving shared interaction structure unmodeled. Consequently, these systems do not jointly represent speaker-grounded states, inter-user relations, and asker-conditioned retrieval over a common history of interactions involving multiple users. Further discussion is provided in Appendix A.1.

6 Conclusion

We presented DG-MEM, a dual-graph framework that extends long-term agent memory from personal to shared conversational settings. By organizing content and user-context structure in complementary graphs with aligned message anchors, DG-MEM combines asker-conditioned retrieval and query-conditioned refinement to recover relevant, attributed evidence. Memory construction and retrieval require no LLM calls. Experiments on GroupMemBench and LoCoMo demonstrate superior overall performance in both multi-user and conventional single-user settings, with consistent advantages across different answering models. Ablation studies support the complementary roles of the two graph views, while efficiency analysis shows substantial reductions in construction time and storage requirements. These findings highlight the importance of jointly modeling conversational content and interaction structure for accurate and efficient memory in shared agents.

Reproducibility Statement

Our code and associated resources are available through the anonymous repository linked in the abstract. The repository includes a detailed README with instructions for reproducing the experiments.

AI use statement

We leveraged AI tools to assist with grammatical revision and language quality improvement. The authors take complete responsibility for all content accuracy and integrity of the final submitted manuscript.

References

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

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory, 2025. URL https://arxiv.org/abs/2504.19413.

DeepSeek-AI. Deepseek-v4.1-flash: Pushing the limits of kv cache compression, 2026. URL https://arxiv.org/abs/2609.19969.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization, 2025. URL https://arxiv.org/abs/2404.16130.

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, Huajun Chen, and Ningyu Zhang. Lightmem: Lightweight and efficient memory-augmented generation. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=dyJ0GWpjJB.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6894–6910, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.552. URL https://aclanthology.org/2021.emnlp-main.552/.

Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=hkujvAPVsg.

Yuyang Hu, Jiongnan Liu, Jiejun Tan, Yutao Zhu, and Zhicheng Dou. Memory matters more: Event-centric memory as a logic map for agent searching and reasoning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 22389–22407, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1123. URL https://aclanthology.org/2026.findings-acl.1123/.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning, 2022. URL https://arxiv.org/abs/2112.09118.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. Memory OS of AI agent. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 25961–25970, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1318. URL https://aclanthology.org/2025.emnlp-main.1318/.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, pp. 39–48, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450380164. doi: 10.1145/3397271.3401075. URL https://doi.org/10.1145/3397271.3401075.

Chris Latimer, Nicoló Boschi, Andrew Neeser, Chris Bartholomew, Gaurav Srivastava, Xuan Wang, and Naren Ramakrishnan. Hindsight is 20/20: Building agent memory that retains, recalls, and reflects, 2025. URL https://arxiv.org/abs/2512.12818.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. Simplemem: Efficient lifelong memory for LLM agents. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=oBgLvd5YC6.

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

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2024. URL https://arxiv.org/abs/2310.08560.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701320. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

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.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of*

the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Jing Xu, Arthur Szlam, and Jason Weston. Beyond goldfish memory: Long-term open-domain conversation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5180–5197, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.356. URL https://aclanthology.org/2022.acl-long.356/.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-mem: Agentic memory for LLM agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=FiM0M8gcct.

B. Y. Yan, Chaofan Li, Hongjin Qian, Shuqi Lu, and Zheng Liu. General agentic memory via deep research, 2025. URL https://arxiv.org/abs/2511.18423.

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

Chang Yang, Chuang Zhou, Yilin Xiao, Su Dong, Luyao Zhuang, Yujing Zhang, Zhu Wang, Zijin Hong, Zheng Yuan, Zhishang Xiang, Shengyuan Chen, Huachi Zhou, Qinggang Zhang, Ninghao Liu, Jinsong Su, Xinrun Wang, Yi Chang, and Xiao Huang. Graph-based agent memory: Taxonomy, techniques, and applications, 2026a. URL https://arxiv.org/abs/2602.05665.

Jingbo Yang, Kwei-Herng Lai, Xiaowen Wang, Shiyu Chang, Yaar Harari, and Evgeniy Gabrilovich. Groupmembench: Benchmarking llm agent memory in multi-party conversations, 2026b. URL https://arxiv.org/abs/2605.14498.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: enhancing large language models with long-term memory. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i17.29946. URL https://doi.org/10.1609/aaai.v38i17.29946.

A Appendix

A.1 Detailed Related Work

Similarity-Based Memory Retrieval. Similarity-based memory represents an interaction history as a collection of retrievable text units. BM25 (Robertson & Zaragoza, 2009) ranks these units using lexical statistics and remains effective when queries share distinctive terms with relevant evidence. Dense retrieval (Lewis et al., 2020; Reimers & Gurevych, 2019; Karpukhin et al., 2020; Izacard et al., 2022; Gao et al., 2021; Khattab & Zaharia, 2020; Chen et al., 2024) instead maps queries and memories into a shared embedding space, enabling relevant content to be recovered under paraphrasing and lexical variation. Hybrid approaches combine sparse and dense signals to improve recall across different query types. These methods (Xu et al., 2022; Park et al., 2023; Zhong et al., 2024) are scalable, inexpensive to construct, and preserve direct access to the original conversation. Their retrieval scores, however, primarily measure content relevance and provide limited information about whose state a retrieved statement describes or how its speaker relates to the asker.

Graph-Structured Memory. Graph-based systems organize memories through entities, events, and their relations. GraphRAG (Edge et al., 2025) extracts entities and relations from source content, partitions the resulting graph into communities, and summarizes those communities to support local and corpus-level information access. HippoRAG (Gutierrez et al., 2024) constructs an entity-centric knowledge graph and propagates query relevance through graph connectivity, allowing evidence associated with multiple entities to be recovered jointly. Zep (Rasmussen et al., 2025) extends structured memory to evolving conversations by recording temporally qualified facts and preserving historical relationships among entities. CompassMem (Hu et al., 2026) segments continuous experience into events and connects them through explicit logical relations, using the resulting event graph to guide evidence collection. These approaches demonstrate the value of relational retrieval, but their effectiveness depends on accurate entity resolution and edge construction. Both operations become more ambiguous in group conversations, where different speakers may discuss overlapping entities, use role-specific terminology, or maintain incompatible states.

Agentic Memory Management. Agentic systems delegate memory construction, revision, or access to an LLM-based controller. MemGPT (Packer et al., 2024) draws on operating-system abstractions and allows an agent to transfer information between its active context and external storage through tool-mediated operations. MemoryOS (Kang et al., 2025) coordinates short-, mid-, and long-term stores through hierarchical updating and retrieval. Mem0 (Chhikara et al., 2025) extracts salient facts and applies explicit operations when new observations reinforce, modify, or supersede stored information. A-Mem (Xu et al., 2025) converts experience into structured notes whose attributes and links evolve as additional evidence becomes available. GAM (Yan et al., 2025) retains lightweight cues during offline processing and employs a researcher agent to reconstruct query-specific context from a complete page store. Hindsight (Latimer et al., 2025) separates accumulated information into world, experience, observation, and opinion networks, coupling temporal and entity-aware recall with preference-conditioned reflection. LightMem (Fang et al., 2026) reduces online overhead through staged compression, topic-aware organization, and offline consolidation. SimpleMem (Liu et al., 2026) combines structured semantic compression and online memory synthesis with intent-aware retrieval planning. Although these systems differ substantially in architecture, many continue to invoke generative models during extraction, consolidation, reflection, or query planning.

Existing approaches across all three directions are predominantly formulated and evaluated around a single memory owner or a fixed conversational dyad. In most implementations, user identifiers define namespace boundaries that separate one history from another. This formulation leaves no explicit representation of the relationships through which users exchange information, revise collective decisions, or maintain simultaneously valid but different beliefs. It also prevents the relevance of evidence from being conditioned jointly on its speaker and the identity of the asker. These missing capabilities distinguish memory over a shared multi-user history from conventional personal conversational memory.

Table 4: Hyperparameter sensitivity of DG-MEM on GroupMemBench. The shaded row denotes the baseline configuration, and bold hyperparameter values indicate changes from it. Best accuracies are **bold** and second-best underlined. Δ denotes the accuracy change from the baseline in percentage points, computed from unrounded accuracies.

`memory_weight`	`ppr_damping`	`candidate_pool`	**Accuracy (%)**	**Δ (pp)**
0.50	0.85	64	**55.30**	—
0.40	0.85	64	54.50	−0.80
0.60	0.85	64	52.35	−2.95
0.50	**0.80**	64	52.75	−2.55
0.50	**0.90**	64	52.35	−2.95
0.50	0.85	**48**	52.88	−2.42
0.50	0.85	**96**	54.50	−0.80

A.2 HYPERPARAMETER ANALYSIS

We examine the sensitivity of DG-MEM to three hyperparameters: `memory_weight`, `ppr_damping`, and `candidate_pool`. These parameters respectively control the balance between the Memory Graph and User-Context Graph, the extent of structural information propagation, and the number of candidates considered during reranking. Starting from the default configuration of $(0.50, 0.85, 64)$, we vary one hyperparameter at a time while keeping the others fixed. As shown in Table 4, accuracy remains between 52.35% and 55.30% across all tested configurations, with a maximum decrease of 2.95 percentage points. Varying `memory_weight` from 0.40 to 0.60 preserves strong performance, with the lower setting remaining within 0.80 points of the default. Changing `ppr_damping` to 0.80 or 0.90 produces decreases of 2.55 and 2.95 points, respectively, showing that effective retrieval is maintained across the tested propagation settings. Performance is also stable over a twofold range of candidate budgets: reducing `candidate_pool` to 48 incurs a 2.42-point decrease, while increasing it to 96 yields an accuracy of 54.50%, only 0.80 points below the default. The performance advantage over existing methods persists throughout these variations. Even the lowest-scoring configuration exceeds Hindsight, the strongest baseline in Table 1, by 6.34 percentage points. These results demonstrate that the benefits of DG-MEM extend across multiple configurations, supporting its robustness to changes in evidence weighting, graph propagation, and candidate selection within the evaluated ranges.

A.3 PER-DOMAIN RESULTS

Table 5: Accuracy (%) of memory systems on the **Finance subset** of GroupMemBench across six query categories. Average denotes overall accuracy. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	**Multi-Hop**	**Update**	**Ambiguity**	**Implicit**	**Temporal**	**Abstention**	**Average**
RAG-based Methods							
text-embedding-3-large	52.10	31.20	22.20	60.00	44.40	72.41	44.40
BM25	47.90 (−4.20)	34.40 (+3.20)	11.10 (−11.10)	53.30 (−6.70)	46.70 (+2.30)	75.90 (+3.50)	42.10 (−2.30)
GraphRAG	22.90 (−29.20)	12.50 (−18.70)	26.70 (+4.50)	6.70 (−53.30)	6.70 (−37.70)	72.41 (0.00)	24.30 (−20.10)
HippoRAG	58.30 (+6.20)	37.50 (+6.30)	40.00 (+17.80)	33.30 (−26.70)	48.89 (+4.49)	69.00 (−3.40)	49.10 (+4.70)
Agent Memory Systems							
Mem0	31.20 (−20.90)	6.20 (−25.00)	8.90 (−13.30)	20.00 (−40.00)	31.10 (−13.30)	86.20 (+13.80)	29.40 (−15.00)
MemGPT	41.67 (−10.43)	18.75 (−12.45)	28.89 (+6.69)	26.67 (−33.33)	17.78 (−26.62)	79.31 (+6.91)	34.58 (−9.82)
A-Mem	56.20 (+4.10)	31.20 (0.00)	33.30 (+11.10)	46.70 (−13.30)	31.10 (−13.30)	69.00 (−3.40)	43.50 (−0.90)
LightMem	8.33 (−43.77)	0.00 (−31.20)	2.22 (−19.98)	6.67 (−53.33)	4.44 (−39.96)	72.41 (0.00)	13.55 (−30.85)
SimpleMem	31.25 (−20.85)	15.63 (−15.57)	17.78 (−4.42)	26.67 (−33.33)	31.11 (−13.29)	51.72 (−20.68)	28.50 (−15.90)
Hindsight	50.00 (−2.10)	31.20 (0.00)	44.40 (+22.20)	33.30 (−26.70)	64.40 (+20.00)	79.31 (+6.91)	51.90 (+7.50)
DG-MEM	64.58 (+12.48)	56.25 (+25.05)	35.56 (+13.36)	60.00 (0.00)	48.89 (+4.49)	79.31 (+6.91)	55.61 (+11.21)

Finance. The Finance subset involves 12 participants spanning seven professional roles, the largest role variety among the four domains. The benchmark's worked example illustrates compliance coordination between Finance and Data Engineering, where answering a question requires associating a specific user's request with the relevant collaborators and project context. In this setting, DG-MEM achieves the highest overall accuracy of 55.61% (Table 5), outperforming Hindsight by 3.71 percentage points. Its advantages are particularly pronounced on Multi-Hop and Update, reaching 64.58% and 56.25%, respectively, with improvements of 6.28 and 18.75 points over the strongest baseline in each category. These gains demonstrate improved performance on questions requiring evidence integration and the identification of updated information within multi-user histories. Moreover, DG-MEM matches the best Implicit accuracy of 60.00%, supporting its effectiveness when the asker's identity must be resolved from the conversational context. Together, these results establish its advantages in financial workplace conversations involving role-specific requests and evolving collaborative decisions.

Table 6: Accuracy (%) of memory systems on the **Technology subset** of GroupMemBench across six query categories. Average denotes overall accuracy. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	31.70	25.00	20.90	46.40	32.40	82.10	37.10
BM25	43.90 (+12.20)	30.60 (+5.60)	23.30 (+2.40)	35.70 (−10.70)	73.00 (+40.60)	78.60 (−3.50)	46.00 (+8.90)
GraphRAG	19.50 (−12.20)	19.40 (−5.60)	16.30 (−4.60)	17.90 (−28.50)	5.40 (−27.00)	78.60 (−3.50)	23.90 (−13.20)
HippoRAG	26.80 (−4.90)	33.30 (+8.30)	27.90 (+7.00)	50.00 (+3.60)	29.70 (−2.70)	78.60 (−3.50)	38.50 (+1.40)
Agent Memory Systems							
Mem0	7.30 (−24.40)	0.00 (−25.00)	11.60 (−9.30)	17.90 (−28.50)	24.30 (−8.10)	89.30 (+7.20)	22.10 (−15.00)
MemGPT	24.39 (−7.31)	25.00 (0.00)	16.28 (−4.62)	35.71 (−10.69)	16.22 (−16.18)	75.00 (−7.10)	29.58 (−7.52)
A-Mem	36.60 (+4.90)	33.30 (+8.30)	25.60 (+4.70)	53.60 (+7.20)	24.30 (−8.10)	67.90 (−14.20)	38.00 (+0.90)
LightMem	0.00 (−31.70)	0.00 (−25.00)	2.33 (−18.57)	7.14 (−39.26)	5.41 (−26.99)	64.29 (−17.81)	10.80 (−26.30)
SimpleMem	24.39 (−7.31)	5.56 (−19.44)	6.98 (−13.92)	21.43 (−24.97)	35.14 (+2.74)	39.29 (−42.81)	21.13 (−15.97)
Hindsight	36.60 (+4.90)	11.10 (−13.90)	37.20 (+16.30)	50.00 (+3.60)	54.10 (+21.70)	71.40 (−10.70)	41.80 (+4.70)
DG-MEM	53.66 (+21.96)	52.78 (+27.78)	23.26 (+2.36)	57.14 (+10.74)	67.57 (+35.17)	85.71 (+3.61)	54.46 (+17.36)

Technology. The Technology subset contains the largest participant pool, with 18 users across six roles, seven projects, and 35 topics. This setting combines diverse speakers with multiple project threads, placing particular demands on retrieving evidence in the appropriate user and conversational context. As shown in Table 6, DG-MEM achieves 54.46% overall accuracy, exceeding the strongest baseline, BM25, by 8.46 percentage points and the strongest agent memory system, Hindsight, by 12.66 points. It leads on Multi-Hop, Update, and Implicit, with accuracies of 53.66%, 52.78%, and 57.14%, respectively. The improvements on Multi-Hop and Update reach 9.76 and 19.48 points over their respective strongest baselines, demonstrating substantial gains in connecting distributed evidence and identifying relevant updates. DG-MEM additionally ranks second on Temporal and Abstention, placing first or second in five of the six categories. This broad performance advantage supports its applicability to multi-user memory settings with many participants and intertwined project discussions.

Healthcare. The Healthcare subset combines a relatively small group of six participants across four roles with ten projects and 79 phases. It also contains 113 decision changes, the highest count among the four domains, making evolving project context an important characteristic even with fewer speakers. Table 7 shows that DG-MEM achieves the best overall accuracy of 49.66%, surpassing Hindsight by 4.26 percentage points. In particular, it obtains the highest Update accuracy of 23.53%, improving over the strongest baselines by 5.93 points. This advantage is relevant to a corpus with repeated decision revisions, where earlier statements must be interpreted alongside subsequent changes. DG-MEM also achieves the highest Abstention accuracy of 94.12%, a gain of 4.62 points over the best baseline, demonstrating improved handling of questions without sufficient supporting information. Across all six categories, it ranks first or jointly first in four and second in the remaining two. These results show that its benefits extend to smaller teams managing multiple projects and changing information.

Table 7: Accuracy (%) of memory systems on the **Healthcare subset** of GroupMemBench across six query categories. Average denotes overall accuracy. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	27.10	17.60	14.29	50.00	10.80	68.40	26.90
BM25	37.50 (+10.40)	11.80 (−5.80)	0.00 (−14.29)	100.00 (+50.00)	45.90 (+35.10)	89.50 (+21.10)	43.10 (+16.20)
GraphRAG	2.10 (−25.00)	5.90 (−11.70)	0.00 (−14.29)	50.00 (0.00)	2.70 (−8.10)	57.90 (−10.50)	11.50 (−15.40)
HippoRAG	35.40 (+8.30)	17.60 (0.00)	0.00 (−14.29)	100.00 (+50.00)	27.00 (+16.20)	73.70 (+5.30)	35.40 (+8.50)
Agent Memory Systems							
Mem0	12.50 (−14.60)	5.90 (−11.70)	0.00 (−14.29)	50.00 (0.00)	0.00 (−10.80)	73.70 (+5.30)	16.90 (−10.00)
MemGPT	12.50 (−14.60)	17.60 (0.00)	0.00 (−14.29)	0.00 (−50.00)	5.60 (−5.20)	89.50 (+21.10)	21.70 (−5.20)
A-Mem	22.90 (−4.20)	11.80 (−5.80)	0.00 (−14.29)	50.00 (0.00)	10.80 (0.00)	68.40 (0.00)	23.80 (−3.10)
LightMem	4.17 (−22.93)	0.00 (−17.60)	0.00 (−14.29)	0.00 (−50.00)	2.70 (−8.10)	88.24 (+19.84)	22.76 (−4.14)
SimpleMem	20.83 (−6.27)	5.88 (−11.72)	14.29 (0.00)	0.00 (−50.00)	16.22 (+5.42)	79.41 (+11.01)	31.03 (+4.13)
Hindsight	45.80 (+18.70)	17.60 (0.00)	14.29 (0.00)	50.00 (0.00)	45.90 (+35.10)	78.90 (+10.50)	45.40 (+18.50)
DG-MEM	39.58 (+12.48)	23.53 (+5.93)	14.29 (0.00)	100.00 (+50.00)	37.84 (+27.04)	94.12 (+25.72)	49.66 (+22.76)

Table 8: Accuracy (%) of memory systems on the **Manufacturing subset** of GroupMemBench across six query categories. Average denotes overall accuracy. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	33.30	13.60	27.30	0.00	39.50	75.80	39.90
BM25	31.10 (−2.20)	13.60 (0.00)	0.00 (−27.30)	0.00 (0.00)	55.80 (+16.30)	72.70 (−3.10)	41.10 (+1.20)
GraphRAG	4.40 (−28.90)	13.60 (0.00)	18.20 (−9.10)	0.00 (0.00)	7.00 (−32.50)	57.60 (−18.20)	18.40 (−21.50)
HippoRAG	35.60 (+2.30)	9.10 (−4.50)	18.20 (−9.10)	0.00 (0.00)	11.60 (−27.90)	78.80 (+3.00)	32.30 (−7.60)
Agent Memory Systems							
Mem0	35.60 (+2.30)	9.10 (−4.50)	27.30 (0.00)	25.00 (+25.00)	9.30 (−30.20)	78.80 (+3.00)	32.90 (−7.00)
MemGPT	11.10 (−22.20)	4.50 (−9.10)	18.20 (−9.10)	0.00 (0.00)	9.30 (−30.20)	72.70 (−3.10)	22.80 (−17.10)
A-Mem	24.40 (−8.90)	0.00 (−13.60)	18.20 (−9.10)	0.00 (0.00)	25.60 (−13.90)	66.70 (−9.10)	29.10 (−10.80)
LightMem	6.67 (−26.63)	0.00 (−13.60)	9.09 (−18.21)	0.00 (0.00)	0.00 (−39.50)	75.00 (−0.80)	23.12 (−16.78)
SimpleMem	26.67 (−6.63)	4.55 (−9.05)	9.09 (−18.21)	0.00 (0.00)	27.91 (−11.59)	77.08 (+1.28)	36.42 (−3.48)
Hindsight	35.60 (+2.30)	9.10 (−4.50)	27.30 (0.00)	0.00 (0.00)	53.50 (+14.00)	78.80 (+3.00)	44.30 (+4.40)
DG-MEM	51.11 (+17.81)	22.73 (+9.13)	63.64 (+36.34)	25.00 (+25.00)	58.14 (+18.64)	91.67 (+15.87)	60.69 (+20.79)

Manufacturing. The Manufacturing subset brings together production managers, quality engineers, maintenance engineers, supply-chain managers, and project managers across ten projects. These distinct professional roles make speaker context and role-dependent terminology salient aspects of cross-functional communication. DG-MEM achieves 60.69% overall accuracy (Table 8), outperforming Hindsight by 16.39 percentage points, its largest overall margin over the strongest baseline across the four domains. It leads outright in five categories and shares the best result in Implicit. The most pronounced improvement appears in Ambiguity, where accuracy reaches 63.64%, exceeding the strongest baselines by 36.34 points. This result is particularly relevant to the benchmark's ambiguity questions, which require interpreting terms according to the asker's context and connecting evidence expressed in different role-specific language. DG-MEM also improves Multi-Hop and Update by 15.51 and 9.13 points, respectively, while achieving the highest Temporal and Abstention accuracies. These gains are consistent with the motivation for jointly modeling memory content and user context in cross-functional group conversations.

A.4 DETAILED RESULTS OF DIFFERENT LLMS

Qwen3-8B. With the smaller Qwen3-8B model, DG-MEM achieves the highest overall accuracy of 37.72% (Table 9), surpassing BM25 by 3.36 percentage points and Hindsight by 4.97 points. Its advantages span several distinct memory requirements. On Multi-Hop and Update, DG-MEM reaches 43.41% and 36.45%, improving over the strongest baselines by 6.60 and 8.41 points, respec-

Table 9: Accuracy (%) of memory systems on GroupMemBench across six query categories, using **Qwen3-8B** as the base LLM. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	29.12	22.43	22.64	42.86	17.28	33.09	26.31
BM25	36.81 (+7.69)	28.04 (+5.61)	18.87 (−3.77)	36.73 (−6.13)	38.89 (+21.61)	41.73 (+8.64)	34.36 (+8.05)
Agent Memory Systems							
MemGPT	20.33 (−8.79)	14.95 (−7.48)	20.75 (−1.89)	32.65 (−10.21)	10.49 (−6.79)	35.97 (+2.88)	21.21 (−5.10)
LightMem	16.48 (−12.64)	3.74 (−18.69)	15.09 (−7.55)	14.29 (−28.57)	13.58 (−3.70)	40.29 (+7.20)	18.12 (−8.19)
SimpleMem	25.27 (−3.85)	14.02 (−8.41)	13.21 (−9.43)	14.29 (−28.57)	29.01 (+11.73)	46.04 (+12.95)	25.91 (−0.40)
Hindsight	34.07 (+4.95)	11.21 (−11.22)	28.30 (+5.66)	48.98 (+6.12)	38.27 (+20.99)	38.85 (+5.76)	32.75 (+6.44)
DG-MEM	43.41 (+14.29)	36.45 (+14.02)	31.13 (+8.49)	44.90 (+2.04)	38.89 (+21.61)	32.37 (−0.72)	37.72 (+11.41)

tively. It also achieves the best Ambiguity accuracy of 31.13%, exceeding Hindsight by 2.83 points, while matching the highest Temporal accuracy of 38.89%. Together with its second-place result on Implicit, DG-MEM ranks first or jointly first in four categories and second in another. These results demonstrate that its benefits remain evident with an 8B-parameter base model, covering evidence integration, user-specific updates, and context-dependent interpretation. The improvements across these complementary categories support the effectiveness of the dual-graph framework at a relatively small model scale.

Table 10: Accuracy (%) of memory systems on GroupMemBench across six query categories, using **Qwen3-32B** as the base LLM. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	35.71	23.36	26.42	40.82	27.78	67.63	37.18
BM25	40.66 (+4.95)	22.43 (−0.93)	14.15 (−12.27)	36.73 (−4.09)	46.30 (+18.52)	78.42 (+10.79)	42.28 (+5.10)
Agent Memory Systems							
MemGPT	23.63 (−12.08)	17.76 (−5.60)	28.30 (+1.88)	40.82 (0.00)	25.93 (−1.85)	73.38 (+5.75)	34.36 (−2.82)
LightMem	18.13 (−17.58)	4.67 (−18.69)	19.81 (−6.61)	18.37 (−22.45)	28.40 (+0.62)	69.06 (+1.43)	28.19 (−8.99)
SimpleMem	29.67 (−6.04)	9.35 (−14.01)	22.64 (−3.78)	34.69 (−6.13)	32.72 (+4.94)	71.94 (+4.31)	34.63 (−2.55)
Hindsight	49.45 (+13.74)	11.21 (−12.15)	33.02 (+6.60)	48.98 (+8.16)	52.47 (+24.69)	69.06 (+1.43)	45.91 (+8.73)
DG-MEM	48.90 (+13.19)	34.58 (+11.22)	32.08 (+5.66)	42.86 (+2.04)	49.38 (+21.60)	67.63 (0.00)	47.65 (+10.47)

Qwen3-32B. Qwen3-32B evaluates whether the advantages of DG-MEM persist with a larger dense model from the same family. As shown in Table 10, DG-MEM retains the highest overall accuracy of 47.65%, outperforming Hindsight by 1.74 percentage points and BM25 by 5.37 points. Its overall accuracy increases by 9.93 points relative to the Qwen3-8B configuration, while its leading position among the evaluated memory systems remains unchanged. The clearest category-level advantage appears on Update, where DG-MEM achieves 34.58%, exceeding the strongest baseline by 11.22 points. This result is particularly relevant to multi-user memory, where identifying an update requires maintaining its association with the appropriate user's history. In addition, DG-MEM ranks second on Multi-Hop, Ambiguity, Implicit, and Temporal, placing among the top two methods in five categories. The combination of a substantial Update advantage and strong performance across the other content-bearing categories supports the framework's effectiveness at both evaluated Qwen3 model scales.

GPT-4o-mini. GPT-4o-mini provides a complementary evaluation with a compact model designed for cost-efficient inference. In this configuration, DG-MEM achieves the best overall accuracy of 42.82% (Table 11), improving over Hindsight and BM25 by 4.03 and 5.10 percentage points, respectively. It leads outright in four categories: Multi-Hop, Update, Ambiguity, and Implicit, with accuracies of 47.25%, 21.50%, 35.85%, and 44.90%. The gains on Multi-Hop and Update reach 3.84

Table 11: Accuracy (%) of memory systems on GroupMemBench across six query categories, using **GPT-4o-mini** as the base LLM. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	32.42	12.15	23.58	34.69	25.31	66.91	33.29
BM25	36.26 (+3.84)	15.89 (+3.74)	14.15 (−9.43)	42.86 (+8.17)	41.36 (+16.05)	68.35 (+1.44)	37.72 (+4.43)
Agent Memory Systems							
MemGPT	24.18 (−8.24)	10.28 (−1.87)	23.58 (0.00)	30.61 (−4.08)	14.20 (−11.11)	60.43 (−6.48)	27.11 (−6.18)
LightMem	6.04 (−26.38)	0.00 (−12.15)	5.66 (−17.92)	6.12 (−28.57)	3.70 (−21.61)	72.66 (+5.75)	17.05 (−16.24)
SimpleMem	23.63 (−8.79)	7.48 (−4.67)	16.98 (−6.60)	20.41 (−14.28)	24.69 (−0.62)	69.78 (+2.87)	28.99 (−4.30)
Hindsight	43.41 (+10.99)	10.28 (−1.87)	33.96 (+10.38)	38.78 (+4.09)	44.44 (+19.13)	51.80 (−15.11)	38.79 (+5.50)
DG-MEM	47.25 (+14.83)	21.50 (+9.35)	35.85 (+12.27)	44.90 (+10.21)	43.83 (+18.52)	56.83 (−10.08)	42.82 (+9.53)

and 5.61 points over the strongest baseline in each category. Its simultaneous lead on Ambiguity and Implicit further extends this advantage to questions requiring context-dependent interpretation and asker-specific reasoning. DG-MEM also ranks second on Temporal with 43.83% accuracy, yielding a top-two position in all five content-bearing categories. This breadth of improvement shows that the overall gain is accompanied by advantages across multiple forms of memory-grounded reasoning, supporting the applicability of DG-MEM with a compact base LLM.

Table 12: Accuracy (%) of memory systems on GroupMemBench across six query categories, using **DeepSeek-V4-Flash** as the base LLM. Parenthesized values are differences from text-embedding-3-large in percentage points, computed from the displayed accuracies. Green/red indicates improvement/degradation. Gray indicates no change.

Method	Multi-Hop	Update	Ambiguity	Implicit	Temporal	Abstention	Average
RAG-based Methods							
text-embedding-3-large	31.87	21.50	16.98	46.94	27.78	83.45	37.99
BM25	40.66 (+8.79)	21.50 (0.00)	14.15 (−2.83)	44.90 (−2.04)	50.00 (+22.22)	88.49 (+5.04)	45.37 (+7.38)
Agent Memory Systems							
MemGPT	26.37 (−5.50)	21.50 (0.00)	17.92 (+0.94)	38.78 (−8.16)	20.37 (−7.41)	87.05 (+3.60)	35.30 (−2.69)
LightMem	21.98 (−9.89)	4.67 (−16.83)	13.21 (−3.77)	10.20 (−36.74)	29.01 (+1.23)	92.09 (+8.64)	32.08 (−5.91)
SimpleMem	31.87 (0.00)	18.69 (−2.81)	20.75 (+3.77)	36.73 (−10.21)	25.31 (−2.47)	86.33 (+2.88)	37.45 (−0.54)
Hindsight	53.30 (+21.43)	13.08 (−8.42)	39.62 (+22.64)	51.02 (+4.08)	61.11 (+33.33)	74.10 (−9.35)	51.01 (+13.02)
DG-MEM	54.40 (+22.53)	37.38 (+15.88)	30.19 (+13.21)	59.18 (+12.24)	54.32 (+26.54)	82.01 (−1.44)	53.96 (+15.97)

DeepSeek-V4-Flash. DeepSeek-V4-Flash extends the evaluation to a mixture-of-experts model family and yields the highest overall accuracy for DG-MEM among the four tested configurations. As reported in Table 12, DG-MEM reaches 53.96%, surpassing Hindsight by 2.95 percentage points and BM25 by 8.59 points. The largest category-level advantage occurs on Update: DG-MEM achieves 37.38%, compared with 21.50% for the strongest baselines, a gain of 15.88 points. It also leads on Implicit with 59.18%, outperforming Hindsight by 8.16 points, and obtains the highest Multi-Hop accuracy of 54.40%. These results highlight improvements in identifying user-specific changes, resolving asker-dependent questions, and combining evidence across the conversation history. Its second-place results on Ambiguity and Temporal further establish broad competitiveness across content-bearing tasks. Together with the Qwen3 and GPT-4o-mini results, this configuration demonstrates a consistent overall advantage across the evaluated model families, with Update remaining a particularly strong category under every base LLM.

A.5 IMPLEMENTATION DETAILS

We implement DG-MEM using standard local encoders and sparse graph operations. No task-specific model is trained, and memory construction and retrieval require no generative-LLM calls. A single retrieval configuration is used across all four GroupMemBench domains.

Preprocessing and local models. For GroupMemBench, each message forms one memory unit together with its available author, role, channel, phase, topic, timestamp, and reply metadata. We use spaCy for named-entity recognition and `all-MiniLM-L6-v2` for message and entity embeddings. For LoCoMo, conversations are converted into overlapping multi-turn windows, and BGE-M3 is used as the embedding model with a lightweight regex-based entity extractor. Candidate reranking uses `cross-encoder/ms-marco-MiniLM-L-6-v2` on both benchmarks.

Graph instantiation. The Content-Centric Memory Graph follows the structure defined in Section 3: message nodes are connected to extracted entity nodes, reply parents, chronologically subsequent messages in the same channel, and adjacent messages in the same interaction phase. Temporal edges decay exponentially with a seven-day half-life. The User-Context Graph connects the aligned message nodes with their authors, roles, conversational scopes, reply parents, and explicitly mentioned users. Conversational scopes are represented by $(\text{channel}, \text{phase}, \text{topic})$, and repeated user participation is sublinearly normalized to prevent highly active users from dominating propagation. Both graphs use Personalized PageRank with damping factor 0.85, at most 40 iterations, and convergence tolerance 10^{-7}. Within the Content-Centric Memory Graph, PPR, dense-similarity, and content-BM25 scores are min–max normalized and fused using $\boldsymbol{\omega}^M = (0.50, 0.25, 0.25)^\top$ in that order. Across the tested configurations of $\boldsymbol{\omega}^M$, our method consistently maintains accuracy above 50% on GroupMemBench, supporting its robustness to the choice of fusion weights.

Cross-graph retrieval and reranking. The Content-Centric Memory Graph first produces a candidate pool by combining dense retrieval, content BM25, and graph propagation. Its top 64 messages seed the corresponding User-Context Graph anchors with reciprocal-rank weight $1/(r+1)$, while the top 64 context-BM25 results use $0.8/(r+1)$. For a self-referential query, the asker node receives weight 1.125; otherwise, it receives weight 0.15. Query-matched role and scope nodes use coefficient 0.60, with at most eight scope nodes activated. We use a candidate pool of 64 messages in both graph stages.

The final ranking combines the content-graph score, context-aware lexical retrieval, user-context propagation, and local cross-encoder relevance. Lightweight query cues only rescale these fixed feature groups; they neither construct additional memories nor invoke a learned or generative controller. Exact feature coefficients, query-pattern definitions, and preprocessing settings are included in the released configuration and source code for reproducibility.

A.6 Case study

Table 13 compares DG-Mem with BM25 and the standalone Content-Centric Memory Graph on a representative Finance question. All methods use the same Top-10 retrieval budget and reader, while the table displays their Top-3 passages for readability. The question requires identifying the current owner-and-expiry tracking policy after an explicit decision update, rather than retrieving any passage that merely mentions ownership, monitoring, or regulatory coverage.

BM25 retrieves lexically related discussions, and the Memory Graph returns semantically similar policies from different projects or earlier decision states. However, neither baseline retrieves the decisive update, Msg_3792, and both consequently abstain. In contrast, DG-Mem ranks Msg_3792 first and correctly recovers the complete updated policy. This example illustrates the complementary roles of the two graphs. Their fusion helps suppress plausible but obsolete or cross-project evidence, which is especially important for knowledge-update and term-ambiguity questions.

A.7 Details of the baselines

We provide detailed descriptions of the baselines, following the two groups introduced in the main text. These methods cover lexical and semantic retrieval, graph-structured indexing, and persistent memory management, providing complementary comparisons for memory-grounded question answering over conversational histories.

RAG-based methods.

- **BM25**. BM25 is a sparse retrieval method that scores textual units through query-term matching, combining inverse document frequency, term-frequency saturation, and document-length

Question	What is the current approach for owner and expiry tracking in the transaction monitoring framework?
Ground Truth	Expand owner and expiry tracking to all register items; low-risk entries remain provisional and are reviewed in the same cutoff cycle.
Gold Evidence	**Msg_3792:** "**Change to the original decision:** owner + expiry tracking now applies to **all register items**; low-risk entries stay **provisional**, but they're reviewed in the same cutoff cycle, not parked separately."
BM25	***Retrieved Top-3 evidence:*** 1) ✗ **Msg_17797:** "The incomplete regulatory coverage gap is still open ... confirm impact by 7/16." *(same framework, no policy update)* 2) ✗ **Msg_4317:** "Leadership needs to decide now whether we freeze the current design or rework it." *(pre-decision request)* 3) ✗ **Msg_1216:** "A blocker on sign-off for triggers, thresholds, and escalation rules." *(related subproblem)* **Coverage:** 0/1—Msg_3792 is missed. **Prediction:** ✗ No information available. **Outcome:** Incorrect.
Memory Graph only	***Retrieved Top-3 evidence:*** 1) ✗ **Msg_6856:** "Tag owner and aging timestamp immediately ... show blocker vs pending verification." *(different project/policy)* 2) ✗ **Msg_7299:** "Keep late hashes in pending verification unless source link or owner is missing." *(different policy)* 3) ✗ **Msg_7528:** "Make owner, revisit date, and re-open trigger mandatory." *(different project)* **Coverage:** 0/1—Msg_3792 is not retrieved. **Prediction:** ✗ No information available. **Outcome:** Incorrect.
DG-MEM	***Retrieved Top-3 evidence:*** 1) ✓ **Msg_3792:** "**Change to the original decision:** owner + expiry tracking now applies to **all register items**; low-risk entries stay **provisional**, but they're reviewed in the same cutoff cycle." *(exact update; rank 1)* 2) ✗ **Msg_6856:** "Tag owner and aging timestamp immediately ..." *(cross-project distractor)* 3) ✗ **Msg_28154:** "Compliance should own the register, and Ops should keep the daily row updates current." *(compatible ownership context)* **Coverage:** **1/1**—Msg_3792 is retrieved at rank 1. **Trace:** `memory` = 0.00, `user_context` = 0.624, `context_match` = 1.00. **Prediction:** ✓ Owner and expiry tracking applies to all register items; low-risk entries remain provisional but are reviewed in the same cutoff cycle. **Outcome:** **Correct**.

Table 13: Compact Finance case study of decision-update retrieval. BM25 and the standalone Memory Graph retrieve related but incorrect or obsolete policies and miss the decisive update. DG-MEM retrieves the exact update at rank 1. The decisive message has zero Memory Graph contribution and a perfect context match, illustrating the incremental value of the User-Context Graph.

normalization. Applied to conversational memory, it retrieves historical units containing relevant names, terms, and other explicit lexical cues. It provides a direct lexical retrieval baseline without requiring LLM-based memory extraction or summarization.

- **text-embedding-3-large**. This dense retrieval baseline uses OpenAI's text-embedding-3-large model to encode queries and textual memory units into a shared vector space. Historical units are ranked by their embedding similarity to the query and supplied as supporting context for answer generation. The resulting comparison evaluates semantic matching beyond exact lexical overlap, using vector representations rather than an explicitly constructed graph or an LLM-managed memory store.
- **GraphRAG** (Edge et al., 2025). GraphRAG constructs an LLM-derived graph index to support question answering over document collections. During indexing, it extracts entities and relation-

ships from text chunks, aggregates their descriptions, and applies hierarchical community detection to organize the resulting graph. An LLM then generates summaries of these communities at different levels of abstraction. For global queries, the original framework generates intermediate answers from community summaries and combines them into a final response. GraphRAG provides a comparison based on entity-centered organization and hierarchical summarization of historical information.

- **HippoRAG** (Gutierrez et al., 2024). Inspired by hippocampal memory indexing theory, HippoRAG connects information distributed across passages through a schemaless knowledge graph. An instruction-tuned LLM performs open information extraction to obtain noun-phrase nodes and relation edges, while a retrieval encoder adds synonymy links between semantically similar nodes. At query time, an LLM extracts named entities from the question, which are linked to graph nodes and used to initialize Personalized PageRank. Propagated node scores are aggregated over their source passages to rank supporting evidence, enabling multi-hop retrieval through a single graph-propagation process.

Agent memory systems.

- **Mem0** (Chhikara et al., 2025). Mem0 maintains persistent conversational memory through incremental extraction and updating. Given a new exchange, an LLM extracts salient facts using recent messages and a running conversation summary as context. Each candidate fact is compared with semantically similar stored memories, and an LLM selects an appropriate operation: adding a new memory, updating an existing memory, deleting contradicted information, or leaving the store unchanged. Relevant memories are subsequently retrieved to support response generation. This design represents incremental fact-level memory maintenance across extended interactions.
- **MemGPT** (Packer et al., 2024). MemGPT introduces operating-system-inspired virtual context management, allowing an LLM to move information between its limited main context and external storage. The main context contains system instructions, editable working memory, and a rolling message queue, while external storage maintains conversation records and archival information. Through function calls, the LLM can update working memory, store information, and retrieve previous records. Memory-pressure warnings, queue summarization, and control-flow mechanisms coordinate these operations as the conversation grows. MemGPT therefore represents self-directed memory access and context management over long interaction histories.
- **A-Mem** (Xu et al., 2025). A-Mem organizes experiences as interconnected notes inspired by the Zettelkasten method. Each note retains interaction content and a timestamp, augmented with LLM-generated keywords, tags, and contextual descriptions. Embedding similarity identifies related historical notes, after which an LLM determines meaningful links and may revise the contextual descriptions and attributes of existing memories. At retrieval time, the query is matched against embeddings of the enriched notes to obtain relevant context. This combination of note construction, linking, and memory evolution provides a comparison based on dynamically organized conversational knowledge.
- **Hindsight** (Latimer et al., 2025). Hindsight organizes persistent memory around three operations: retain, recall, and reflect. Its memory architecture distinguishes world facts, agent experiences, opinions, and synthesized observations. During retention, LLM-based extraction produces narrative facts with temporal information and participant attribution, which are connected through entity, temporal, semantic, and causal links. Recall combines semantic, keyword, graph, and temporal retrieval, followed by reciprocal rank fusion and cross-encoder reranking. Reflection integrates retrieved evidence with a configurable behavioral profile to support preference-conditioned reasoning and opinion updates. Hindsight provides a structured memory baseline combining multiple retrieval signals with explicit temporal and belief-related representations.
- **LightMem** (Fang et al., 2026). LightMem targets efficient memory-augmented generation through sensory, short-term, and long-term memory stages inspired by the Atkinson–Shiffrin model. Sensory memory reduces redundant input through pre-compression and organizes the retained content into topic-based segments. Short-term memory groups these segments and uses an LLM to generate concise, topic-aware summaries for long-term storage. Long-term memory separates immediate insertion from subsequent consolidation, using offline sleep-time updates that can be processed in parallel. This architecture provides an efficiency-oriented comparison for processing and maintaining growing conversational histories.

- **GAM** (Yan et al., 2025). General Agentic Memory follows a just-in-time approach to constructing task-specific context. Its Memorizer maintains lightweight memory cues while preserving complete historical sessions in a page-store. At query time, a Researcher iteratively plans searches, retrieves relevant pages, integrates the collected information, and reflects on whether additional evidence is needed. Its search tools include dense retrieval, BM25, and direct page access by identifier. The resulting context combines integrated findings with relevant source information, providing a comparison based on query-time agentic search over preserved interaction histories.
- **MemoryOS** (Kang et al., 2025). MemoryOS manages conversational information through short-term memory, mid-term memory, and long-term personal memory, coordinated by storage, updating, retrieval, and generation modules. Recent exchanges are maintained as dialogue pages and transferred into mid-term storage through a dialogue-chain-based FIFO mechanism. Segmented storage and heat-based updates organize accumulated information and support the maintenance of longer-term personal knowledge. Retrieval first identifies relevant mid-term segments and then selects dialogue pages within them, combining these results with recent context and personal memory. This hierarchical design supports conversational continuity and personalization across sessions.
- **CompassMem** (Hu et al., 2026). CompassMem represents experience as a hierarchical Event Graph that supports active memory search. An LLM segments incoming observations into coherent events, recording their content, temporal information, summaries, and participants, and extracts logical relations such as causality and temporal order. Incremental updates integrate new events and maintain their connections. At query time, a Planner decomposes the information need into subgoals, while Explorer agents navigate complementary graph paths and collect supporting evidence. A Responder synthesizes the retrieved evidence into an answer. CompassMem provides an event-centered comparison for retrieving distributed evidence through explicit logical relations.
- **SimpleMem** (Liu et al., 2026). SimpleMem combines semantic structured compression, online semantic synthesis, and intent-aware retrieval planning. During memory construction, an LLM filters redundant dialogue and converts informative content into self-contained statements, resolving references and normalizing temporal expressions. Related fragments are synthesized within the current session before storage, and the resulting memories are indexed through dense semantic representations, sparse lexical features, and symbolic metadata. At query time, an LLM generates a retrieval plan specifying search queries and retrieval depth across these complementary views. Their results are merged into a compact context, providing a comparison based on compressed memory representations and adaptive evidence retrieval.