Memory Type Varies: Empowering LLM Agents for Long-Term Memory with Diverse Strategies

Yi Wen[1], **Derong Xu**[1*], **Pengyue Jia**[1], **Yichao Wang**[2*], **Yingyi Zhang**[1], **Maolin Wang**[1], **Junyi Li**[1], **Wenlin Zhang**[1], **Xiaopeng Li**[1*], **Yong Liu**[2], **Xiangyu Zhao**[1*]
[1]City University of Hong Kong, [2]Huawei Noah's Ark Lab
yiwen23-c@my.cityu.edu.hk, wangyichao5@huawei.com, xianzhao@cityu.edu.hk

Abstract

The memory capabilities of Large Language Models (LLMs) have garnered increasing attention recently. Despite great success achieved, existing retrieval-based memory approaches typically overlook the differences between memories and employ a unified strategy to process all memories, leading to suboptimal performance. Thus, an intuitive question arises: can we categorize memory into different types and select appropriate strategies? However, given the topic-rich, scenario-complex, and boundary-blurred nature of memory scenarios, achieving precise classification of memories is not easy. To address this challenge, we propose a memory multi-class dataset in this paper, termed **TriMEM**, which provides precise annotations for memory types across diverse scenarios. Building upon this foundation, we propose a novel memory framework, named **MemoType**, which can adaptively recognize each memory and query type with the learned router model. With the memory and query routing, MemoType can retrieve the memory with corresponding query types and design tailored retrieval strategies, thereby enhancing the retrieval performance. Moreover, we theoretically prove that any single retrieval strategy is subject to a fundamental upper bound on its expected query-document alignment in multi-class corpora, leading to systematic performance degradation. Extensive experiments on three datasets demonstrate that MemoType consistently outperforms existing methods, achieving up to 16.18% improvement in Recall@1[2].

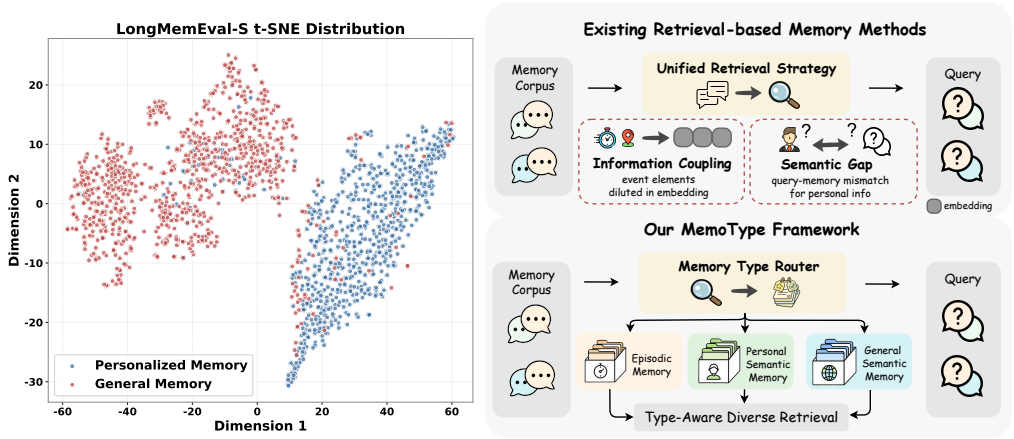
1 Introduction

Large Language Models (LLMs) have demonstrated remarkable abilities in natural language understanding and multi-turn dialogue fields, garnering increasing attention. However, with the growing frequency of User-LLM interactions, the limited context window length of LLMs fails to meet the demand for scalable conversational interactions [Liu et al., 2023]. When presented with retrospective queries, LLMs cannot effectively utilize conversation memories to achieve personalized responses, resulting in the "Forgetting" problem. To address this issue, Retrieval-Augmented Generation (RAG)-based memory techniques have emerged as a practical solution. Instead of concatenating all conversations, RAG-based approaches store conversations in an external bank for selective retrieval, which enables the LLM to recall critical memories and achieve personalized responses.

RAG-based memory strategies [Xu et al., 2025c, Lu et al., 2023] can be broadly classified into unstructured and structured types, based on whether to construct additional memory associative information. Unstructured memory augmentation approaches typically store dialogue fragments or summaries, retrieved through similarity matching with the given query. An early and influential effort is the MemoryBank [Zhong et al., 2024], which maintains a user-specific memory and retrieves facts

*Corresponding authors.
[2]https://github.com/Applied-Machine-Learning-Lab/NeurIPS26_MemoType.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) The t-SNE distribution of LongMemEval-S. (b) Existing unified retrieval vs. MemoType framework.

Figure 1: Evidence and Motivation: Semantic Clustering of memory and the **MemoType** framework.

with relevance scoring to produce appropriate responses. On this basis, SeCom [Pan et al., 2025] employs adaptive memory segmentation techniques to achieve superior performance. Structural memory augmentation strategies organize memories into meaningful relationships, thereby achieving stronger sense-making and richer associations. For instance, HippoRAG [Gutiérrez et al., 2024] builds a graph and uses personalized pageRank to surface associative passages. Structural memory augmentation strategies deepen memory connections while inevitably introducing additional overhead.

Despite great success, the above methods still suffer certain drawbacks. In the retrieval phase, most approaches overlook the structural and content differences between memories and employ a unified strategy to process all memories, leading to suboptimal performance. For instance, event-related memories often contain fragmented information such as persons, time, and location, preventing regular embedding-based similarity from achieving precise retrieval. Conversely, when memories pertain only to basic concepts, introducing details like time or persons may introduce noise into the retrieval process. Thus, an intuitive question arises:

> 💡 *Can we categorize memory into different types and select tailored strategies for each type?*

However, addressing the aforementioned issues presents two major obstacles: i) *Existing memory systems lack a unified classification framework*. Developing a comprehensive scheme that adequately captures the diversity of real conversation scenarios remains a significant challenge.[3]. ii) *Precise classification is inherently difficult given the topic-rich, scenario-complex, and boundary-blurred nature of memory scenarios*. Although some attempts have been conducted, most research [Zhang et al., 2025a, Wang and Chen, 2025] relies on human-crafted categories or adopts basic prompt-based methods to determine memory types, making it difficult to adapt to complex memory scenarios to achieve accurate classification, thus failing to acquire optimal results.

To tackle these challenges, we first draw on Tulving's distinction theory [Tulving et al., 1972] in cognitive psychology [Eysenck and Keane, 2020], which distinguishes memory into episodic and semantic components. Episodic memory records events tied to specific time, place, and participants, whereas semantic memory captures facts and stable knowledge. However, this coarse dichotomy is insufficient for real-world dialogue. Event-centric discussions are relatively rare. For instance, in the LongMemEval-S dataset, episodic memory accounts for only 12.35% of instances, making semantic memory overwhelmingly dominant and poorly resolved. This observation motivates a finer-grained semantic classification. Surprisingly, the t-SNE visualizations of LongMemEval, as shown in Figure 1a, reveal a pronounced semantic separation within semantic memory: the personal content and general knowledge form two distinct clusters with a clear semantic gap. Guided by this evidence, we introduce a well-defined categorization principle consisting of Episodic Memory, Personal Semantic Memory, and General Semantic Memory.

[3]In this paper, the memory we discuss primarily refers to dialogue memory, which mainly encompasses two categories: user-to-user conversations and user-to-assistant conversations. This differs from the experience memory in agent fields. We will not repeat this explanation in subsequent sections.

Inspired by the above classification principle, we construct a memory multi-class dataset, termed **TriMEM**, a memory multi-class dataset comprising 6,000 samples with fine-grained type annotations across diverse topics and scenarios. With the help of TriMEM, we propose a novel memory framework, named **MemoType**. As shown in Figure 1b, MemoType can adaptively recognize each memory and query type with the learned router model. With the memory and query routing, MemoType can retrieve the memory with corresponding query types and design tailored retrieval strategies, thereby enhancing the retrieval in both effectiveness and efficiency. Besides, we design a memory pruning module that adaptively prunes retrieved memories based on queries, reducing irrelevant information. Moreover, we provide a theoretical analysis showing that any single retrieval strategy is subject to a fundamental upper bound on its expected query-document alignment in multi-class corpora, leading to systematic performance degradation. Our contributions are as follows:

- Inspired by Tulving's distinction theory, we propose a novel categorization principle and construct **TriMEM**, a memory multi-class dataset with accurate annotations spanning diverse topics, providing a solid foundation for memory-type classification in memory field.
- We propose **MemoType**, a novel memory framework that adaptively recognizes memory and query types via a learned router, applies tailored retrieval strategies for each type, and further prunes irrelevant information from retrieved memories through a memory pruning module.
- We theoretically prove that any single retrieval strategy has a fundamental upper bound on its expected query-doc alignment in multi-class corpora, leading to systematic performance degradation.

2 MemoType

2.1 PRELIMINARY

Denote $\mathcal{M} = \{\mathcal{S}_i\}_{i=1}^{S}$ as the stored conversation history between two participants, where the dialogue may occur between users or between a user and an assistant. S indicates the number of sessions, $\mathcal{S}_i = \{d_l\}_{l=1}^{S_i}$ denotes the i-th session, comprising S_i sequential dialogue. Denote the retrieval function as f_R and the response generation function as f_G. The research framework process can be described as follows: (1) *Memory Construction*: Build a memory repository $\mathcal{B}$ from $\mathcal{M}$. (2) *Memory Retrieval*: Given a user query q and a retrieval repository B, retrieve n memory turns $\{t_i\}_{i=1}^{n} \subseteq \mathcal{B}$ relevant to q using the function $f_R(q, \mathcal{B}, n)$. (3) *Response Generation*: Employ the retrieved memory turns $\{t_i\}_{i=1}^{n}$ and user query q to generate the output r using $f_G(q, \{t_i\}_{i=1}^{n})$.

2.2 TriMEM Dataset

As mentioned in the previous sections, different types of memories require tailored strategies to optimize performance. A natural idea is to categorize memories into distinct types and apply appropriate strategies accordingly. However, due to the topic-rich, scenario-complex, and boundary-blurred nature of memory scenarios, achieving precise classification is a challenging task. While some preliminary [Wang and Chen, 2025, Zhang et al., 2025a] efforts have been made in memory categorization, most existing approaches rely on basic prompt-based methods to determine memory types, while we verify that prompt-based methods fail to adapt to complex memory scenarios to achieve accurate classification in Table 5 in the Appendix B.5.

Existing memory classification efforts remain limited. The most relevant work, PerLTQA [Du et al., 2024], provides an initial taxonomy but exhibits key shortcomings that limit its practical applicability: it assigns only a single label per memory, overlooking the multi-topic nature of real conversations; its semantic memory category covers only user-related content while neglecting general world knowledge; and it uniformly treats all dialogues as episodic memory, conflating the dialogue form with its content type. These limitations motivate the need for a more comprehensive and fine-grained classification benchmark grounded in real-world dialogue scenarios.

To this end, we introduce **TriMEM**, a memory multi-class dataset of 6,000 samples drawing on Tulving's distinction theory [Eysenck and Keane, 2020]. TriMEM provides fine-grained type annotations: *Episodic Memory* for event-specific recollections, *Personal Semantic Memory* for user-related knowledge, and *General Semantic Memory* for factual world knowledge, across both user-user and user-assistant scenarios. The detailed classification criteria are provided in Table 4, and the dataset statistics are presented in Figure 4 in the Appendix. Due to the space limit, further details about **TriMEM** construction and evaluation can be found in Appendix B.

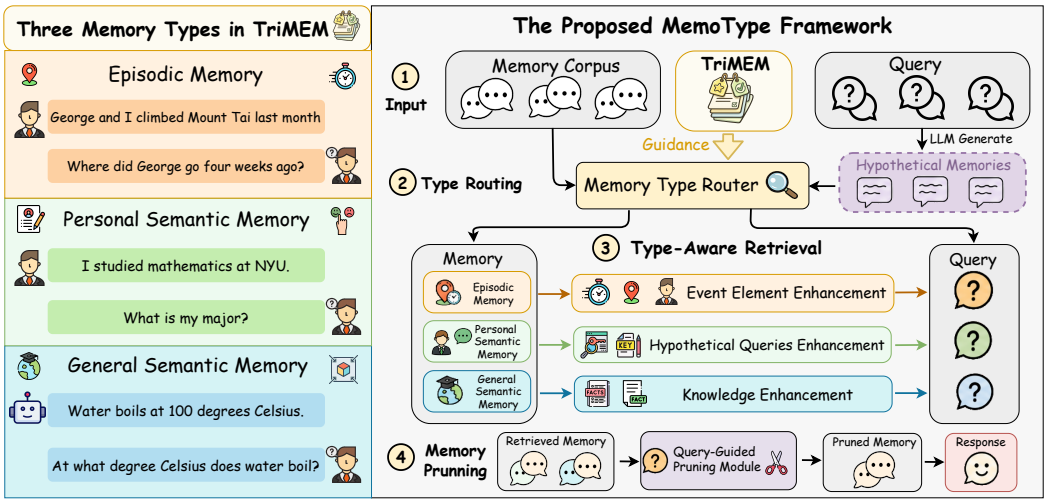


Figure 2: Three Memory Types in **TriMEM** (Left) and the proposed **MemoType** Framework (Right).

2.3 Type Router

Memory Type Router. With the assistance of the **TriMEM** dataset, we fine-tune a lightweight Qwen3-1.7B model with LoRA as the memory type router, formulated as a multi-label classification task to accommodate the multi-topic nature of real-world dialogues. Compared to prompt-based classification with large-scale LLMs, our fine-tuned router achieves superior accuracy (Table 5 in Appendix B.5). Notably, as shown in Table 10 in Appendix D.5, our experiments show that even lightweight encoders such as BERT can achieve competitive routing performance, indicating that the dataset's quality, rather than model scale, is the key enabler for accurate memory classification.

Query Type Router. Following memory classification, a challenge emerges in achieving accurate query routing. Direct query classification is unreliable due to the semantic asymmetry between queries and memories. Queries are typically short and underspecified (e.g., "What is my major?"), while memories contain richer information, making it difficult to classify queries directly into types.

To bridge this gap, we propose to transform queries into the memory space. Specifically, given a query q, we generate K *hypothetical memories* [Gao et al., 2023] that the query might retrieve:

$$\{\tilde{m}_i\}_{i=1}^{K} = \mathcal{G}_{\text{LLM}}(q; \mathcal{P}), \quad \tilde{m}_i \in \mathcal{M}, \tag{1}$$

where $\mathcal{G}_{\text{LLM}}$ denotes the hypothetical memory generation function, $\mathcal{P}$ is the generation prompt, and $\mathcal{M}$ represents the memory space. The generated hypothetical memories $\{\tilde{m}_i\}_{i=1}^{K}$ are subsequently classified by the pre-trained memory router f_ϕ, and the query type is determined as the most frequent category among the K predictions.

Type-Aware Memory Filtering. With the memory router and query type router established, MemoType enables *type-aware retrieval* by constraining the search space based on the predicted types of hypothetical memories. Specifically, let $\mathcal{M} = \bigcup_{c \in \mathcal{C}} \mathcal{M}_c$ denote the full memory corpus partitioned by category. Given the K hypothetical memories $\{\tilde{m}_i\}_{i=1}^{K}$ generated for query q, we first obtain their predicted types $\{c_i = f_\phi(\tilde{m}_i)\}_{i=1}^{K}$ and construct the candidate category set: $\mathcal{C}_q = \{c_i \mid i = 1, \ldots, K\}$. The retrieval is then performed as follows:

$$\mathcal{R}(q) = \text{Retrieve}\left(q, \bigcup_{c \in \mathcal{C}_q} \mathcal{M}_c\right). \tag{2}$$

This filtering strategy significantly reduces the retrieval space, eliminating approximately 50% of unrelated memories across various datasets. We set K=5 to balance coverage and robustness.

2.4 Type-Specific Retrieval Strategies

As previously mentioned, most existing approaches employ a single strategy to process all memories, resulting in suboptimal performance. Therefore, in this subsection, we will adopt different strategies for different memory types to enhance the final performance.

Episodic Memory. Episodic memories are inherently compositional, encoding structured elements such as temporal information, participants, and other event elements. When encoded as monolithic units, these heterogeneous elements are compressed into a single embedding vector, causing fine-grained cues to be overshadowed by the dominant event semantics. This observation motivates a decompose-then-align strategy: rather than matching at the holistic level, we explicitly decompose both queries and memories into typed elements and compute element-wise alignment.

Formally, let $\mathcal{N} = \{T, P, L, E\}$ denote the temporal information, participants, location, and event context, respectively. For a query q, the LLM extracts elements $\mathcal{E}_q = \{e_t\}_{t\in\mathcal{N}}$. The relevance score between query q and the i-th memory m^i is then computed as:

$$s_{\text{EM}}(q, m^i) = \text{sim}(q, m^i) + \sum_{t\in\mathcal{N}} \text{sim}(e_t, m^i_t), \tag{3}$$

where m^i_t is the event element extracted from the i-th memory m^i.

We note that episodic memory, which often requires temporal reasoning, constitutes the most challenging category among the three types. While element decomposition introduces additional computational cost, the routing mechanism (§2.3) restricts this strategy to the episodic subset, avoiding the corpus-wide structural expansion, which substantially amortizes the per-query overhead.

Personal Semantic Memory. The core challenge of PM retrieval lies in the semantic asymmetry between queries and memories. Queries targeting personal knowledge tend to be underspecified (e.g., "What is your occupation?"), while the corresponding memories contain substantially richer context. This asymmetry makes it difficult to generate faithful hypothetical memories from the query side.

We therefore reverse the expansion direction: instead of enriching the query, we enrich each memory m_i by generating K hypothetical queries $\{\tilde{q}_j\}_{j=1}^{K} = \mathcal{G}_{\text{LLM}}(m_i)$ that the memory could plausibly answer. Since the memory carries richer context, the generated questions are more faithful and diverse. The relevance between query q and memory m_i is measured as:

$$s_{\text{PM}}(q, m_i) = \max_{j\in\{1,\ldots,K\}} \text{sim}(q, \tilde{q}_j). \tag{4}$$

The final ranking fuses s_{PM} with the original query-memory similarity via Reciprocal Rank Fusion.

General Semantic Memory. Unlike the preceding two types, general semantic memory primarily encompasses factual world knowledge whose expression is relatively standardized (e.g., "Water boils at 100°C"). This regularity enables LLMs to generate high-quality hypothetical memories that closely approximate the actual memory content, making query-side expansion effective without additional structural decomposition or direction reversal. We therefore adopt the hypothetical memory strategy and aggregate the retrieval results from both the original query and the hypothetical memory via Reciprocal Rank Fusion [Rackauckas, 2024].

2.5 Memory Pruning Module

Retrieved memory segments are typically multi-turn dialogue excerpts that span multiple topics within a single segment. As a result, even when a segment is correctly retrieved, it often contains substantial content unrelated to the query, which may mislead the generator and degrade response quality. This issue is more pronounced in dialogue memory than in standard document retrieval, where passages tend to be topically coherent.

To mitigate this, we introduce a *Query-Conditioned Memory Pruning* module that filters retrieved content before generation. Let $\mathcal{M}_q = \{m_1, m_2, \ldots, m_n\}$ denote the retrieved memory segments for query q. The pruning operation selects a relevant subset via query-conditioned assessment:

$$\mathcal{M}^p_q = \big\{m_i \in \mathcal{M}_q \mid \mathcal{J}_{\text{LLM}}(m_i, q) = 1\big\}, \tag{5}$$

where $\mathcal{J}_{\text{LLM}}(m_i, q)$ is a binary relevance judgment, indicating whether memory m_i contains information pertinent to q. The pruned memory $\mathcal{M}^p_q$ is then utilized for downstream response generation.

3 Theoretical Analysis

In this section, we provide a theoretical analysis demonstrating that single embedding retrieval methods face limitations when the corpus exhibits semantic heterogeneity.

Problem Setup. Consider a corpus $\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$ where documents are drawn from C distinct latent distributions $\mathcal{D}_1, \ldots, \mathcal{D}_C$, with categories $S_1, \ldots, S_C$ respectively. In this section, we consider a unified embedding-based retrieval strategy, where a shared encoder ϕ maps queries and documents to a d-dimensional space, and documents are ranked by inner product similarity $\langle \phi(q), \phi(d) \rangle$. Without loss of generality, we assume all embeddings $\phi(\cdot)$ are ℓ_2-normalized, as retrieval depends only on the relative ranking. For category S_i, we denote its embedding centroid as $\mu_i = \mathbb{E}_{d \sim \mathcal{D}_i}[\phi(d)]$ and its intra-category standard deviation as σ_i.

To formalize corpus heterogeneity, we define the angular separation between categories S_i and S_j as:

$$\delta_{ij} = 1 - \langle v_i^*, v_j^* \rangle,$$

where $v_i^* = \mu_i / \|\mu_i\|$ is the direction of S_i's centroid.

Assumption 3.1. There exists a constant $\delta > 0$ such that $\delta_{ij} \geq \delta$ satisfies for all distinct category pairs $i \neq j$.

Remark 3.2. This assumption requires only that different categories have non-identical centroid directions. When categories are semantically distinct or form clusters, this condition is naturally satisfied. Figure 1a and Figure 13 in the Appendix validate the rationality of the assumption.

Theorem 3.3 (Upper Bound for Single Retrieval)**.** *Let $\mathcal{D}$ satisfy Assumption 3.1 with angular separation $\delta > 0$. Assume documents are uniformly distributed across C categories. Then, for any retrieval encoder ϕ, the expected query-document similarity has an upper bound:*

$$\mathbb{E}_{q \sim \mathcal{Q}, d \sim \mathcal{D}}[\langle e_{query}, e_{doc} \rangle] \leq 1 - \frac{(C-1)\delta}{2C},$$

where $e_{query} = \phi(q)/\|\phi(q)\|$ and $e_{doc} = \phi(d)/\|\phi(d)\|$.

Remark 3.4. The theorem reveals that the expected query-document alignment under a shared encoder is fundamentally constrained by two factors: the number of categories C and the inter-category separation δ. As category diversity increases or categories become more separated, this alignment ceiling tightens.

Remark 3.5. The bound in Theorem 3.3 implies that improvements in alignment for one category necessarily come at the cost of suppressing alignment for others. For these underserved categories, the resulting low similarity scores push their relevant documents to lower positions in the ranking over the heterogeneous corpus, thus degrading retrieval precision. Thus, *an inherent limitation emerges: a single retrieval strategy cannot excel at all categories simultaneously*, which motivates the type-aware design in Section 2.4.

4 Experiment

4.1 Experimental Settings

Dataset. In the section, we evaluate our model performance on four long-term memory datasets: LongMemEval-S [Wu et al., 2025], LoCoMo [Maharana et al., 2024], LongMemEval-M [Wu et al., 2025], and PerLTQA [Du et al., 2024]. These datasets are designed to assess both retrieval and generation tasks. Detailed dataset statistics are provided in the Appendix C.1.

Metrics. We evaluate two types of tasks: retrieval tasks and generation tasks, using metrics tailored to each. For retrieval tasks, we use Recall@k and NDCG@k. For generation tasks, we employ BLEU, BERTScore, F1, Rouge1, Rouge2, RougeLSum, and GPT4Judge [Zheng et al., 2023]. GPT4Judge leverages GPT-5.2 to assess response alignment with reference answers.

Baselines. We evaluate the proposed method with nine baselines. (1)*HyDE* [Gao et al., 2023] (2) *Mill* [Jia et al., 2024] (3) *Query2Doc* [Wang et al., 2023a] (4) *Secom* [Pan et al., 2025] (5) *A-Mem* [Xu et al., 2025c] (6) *HippoRAG 2* [Gutiérrez et al., 2025] (7) *Raptor* [Sarthi et al., 2024] (8) *MemoryOS* [Kang et al., 2025] (9) *LightMem* [Fang et al., 2025a]. Due to the space limits, detailed descriptions of baselines are provided in the Appendix C.2.

Implement Details. For all methods, we adopted a consistent strategy to ensure a fair comparison. In both retrieval and generation tasks, our method and the baselines utilized Contriever as the embedding model. For generation performance, we evaluated using the top 3 retrieved turns on the LongMemEval-S and LongMemEval-M datasets, while for LoCoMo, we selected the top 10 retrieved turns for performance assessment. In our experiments, we utilize 'gpt-4o-mini' as backbone for

Table 1: Retrieval Performance. All methods are based on Contriever as the retriever.

Method	Recall@1	NDCG@1	Recall@3	NDCG@3	Recall@5	NDCG@5	Recall@10	NDCG@10
LongMemEval-S								
HyDE [Gao et al., 2023]	27.23	27.23	53.19	37.82	65.32	42.84	82.34	49.83
Mill [Jia et al., 2024]	38.72	38.72	67.66	50.17	78.94	56.10	90.64	61.34
Query2Doc [Wang et al., 2023a]	25.96	25.96	50.64	36.43	64.89	41.66	81.06	48.38
SeCom [Pan et al., 2025]	45.32	45.32	74.04	53.42	83.19	59.42	91.28	64.66
HippoRAG2 [Gutiérrez et al., 2025]	50.64	50.64	80.85	60.59	88.51	66.48	**93.53**	71.05
A-Mem [Xu et al., 2025c]	48.20	48.20	74.80	56.63	82.80	61.52	91.40	66.83
MemoryOS [Kang et al., 2025]	35.70	35.70	61.80	38.30	73.07	43.82	85.39	48.56
LightMem [Fang et al., 2025a]	56.99	56.99	73.28	51.59	79.12	55.57	88.10	60.24
MemoType (Ours)	**59.15**	**59.15**	**83.19**	**66.74**	**90.21**	**71.60**	92.77	**74.85**
LoCoMo								
HyDE [Gao et al., 2023]	25.33	25.33	42.55	36.24	50.45	39.60	59.82	42.65
Mill [Jia et al., 2024]	19.84	19.84	35.60	30.16	42.75	33.10	53.12	36.57
Query2Doc [Wang et al., 2023a]	23.11	23.11	39.78	33.94	47.48	37.15	59.37	41.11
SeCom [Pan et al., 2025]	23.77	23.77	39.38	34.06	45.57	36.60	55.44	39.76
HippoRAG2 [Gutiérrez et al., 2025]	24.97	24.97	39.98	33.84	45.87	36.28	53.17	38.64
A-Mem [Xu et al., 2025c]	28.80	28.80	46.12	39.82	53.32	42.87	63.29	46.14
MemoryOS [Kang et al., 2025]	19.89	19.89	33.79	30.02	40.28	32.82	49.09	35.64
LightMem [Fang et al., 2025a]	22.56	22.56	31.22	26.65	36.30	28.56	41.84	30.25
MemoType (Ours)	**31.07**	**31.07**	**47.73**	**40.97**	**54.08**	**43.66**	**65.01**	**47.12**
LongMemEval-M								
HyDE [Gao et al., 2023]	21.70	21.70	40.43	27.90	50.64	31.99	63.62	36.71
Mill [Jia et al., 2024]	28.51	28.51	52.77	37.48	63.40	42.73	76.81	48.04
Query2Doc [Wang et al., 2023a]	19.15	19.15	37.02	25.66	44.68	28.96	61.49	34.03
SeCom [Pan et al., 2025]	31.91	31.91	55.11	38.16	64.47	42.38	75.96	47.06
MemoryOS [Kang et al., 2025]	28.18	28.18	47.39	30.30	55.95	33.78	67.22	37.41
LightMem [Fang et al., 2025a]	27.14	27.14	31.73	31.16	33.19	31.82	34.45	32.29
MemoType (Ours)	**48.09**	**48.09**	**70.43**	**52.20**	**78.09**	**57.09**	**84.89**	**61.27**
PerLTQA								
HyDE [Gao et al., 2023]	17.09	17.09	56.06	53.16	66.42	58.01	79.49	62.52
Mill [Jia et al., 2024]	50.18	50.18	79.18	76.46	86.22	79.76	92.25	81.85
Query2Doc [Wang et al., 2023a]	26.15	26.15	62.26	59.34	72.90	64.36	83.47	68.04
SeCom [Pan et al., 2025]	49.51	49.51	77.59	75.14	84.81	78.54	91.97	81.02
HippoRAG2 [Gutiérrez et al., 2025]	41.90	41.90	72.16	69.87	78.05	72.64	85.94	75.41
A-Mem [Xu et al., 2025c]	31.61	31.61	63.11	60.83	72.06	65.07	81.08	68.22
MemoryOS [Kang et al., 2025]	57.19	57.19	79.14	77.18	84.78	79.81	91.61	82.18
LightMem [Fang et al., 2025a]	49.68	49.68	75.65	73.25	82.28	76.34	89.15	78.73
MemoType (Ours)	**71.92**	**71.92**	**88.05**	**86.22**	**92.78**	**88.48**	**96.83**	**89.90**

all tasks and baselines, including memory information processing and question-answer generation. To ensure fair comparisons, all baselines are implemented with uniform generation prompts. The temperature of LLMs is fixed at 0 to ensure reproducibility.

4.2 Overall Results

In this paper, we evaluate two types of tasks, retrieval and generation tasks, with their respective results presented in Table 1 and Table 2. Notably, Raptor involves generating a new text process, making it impossible to assess their retrieval performance. Additionally, for A-Mem, HippoRAG2, and Raptor, the memory construction latency on LongMemEval-M exceeds one week, and thus, we do not report their performance in this study.

Retrieval Results. As shown in Table 1, MemoType consistently achieves the best performance across all four datasets. On LongMemEval-S, MemoType achieves a Recall@1 of 59.15%, outperforming the strongest baseline LightMem by 2.16% and the structural method HippoRAG2 by 8.51%. On the more challenging LongMemEval-M dataset, MemoType achieves a Recall@1 of 48.09%, surpassing SeCom by over 16 points, demonstrating the substantial advantage of type-aware retrieval over uniform segmentation-based approaches. On LoCoMo, MemoType outperforms all baselines including A-Mem by 2.27% in Recall@1, confirming its effectiveness on user-user dialogue scenarios. On PerLTQA, MemoType achieves 71.92% Recall@1, exceeding the second-best method MemoryOS by 14.73%. These results validate that leveraging memory type diversity with tailored strategies yields consistent improvements over both query expansion and structural augmentation methods across diverse memory benchmarks.

Question Answer Results. To validate the question answer performance of the proposed method, we conduct the QA experiment in Table 2. MemoType consistently achieves the best performance

Table 2: Question Answer Performance. All methods are based on Contriever as the retriever.

Method	GPT4Judge	F1	BLEU	Rouge1	Rouge2	RougeL	RougeLsum	BERTScore
LongMemEval-S								
HyDE [Gao et al., 2023]	42.60	9.96	1.44	10.58	4.67	9.31	9.57	83.02
Mill [Jia et al., 2024]	42.20	9.91	1.55	10.57	4.54	9.20	9.45	82.96
Query2Doc [Wang et al., 2023a]	43.00	9.80	1.47	10.44	4.46	9.21	9.43	82.97
SeCom [Pan et al., 2025]	44.80	11.01	1.65	11.66	5.14	10.36	10.58	83.26
HippoRAG2 [Gutiérrez et al., 2025]	45.40	10.55	1.65	11.15	5.17	9.98	10.17	83.12
RAPTOR [Sarthi et al., 2024]	32.20	12.08	1.90	12.73	5.82	11.25	11.35	83.50
A-Mem [Xu et al., 2025c]	47.00	12.12	1.90	12.81	5.91	11.38	11.63	83.58
MemoryOS [Kang et al., 2025]	43.40	13.50	2.09	14.16	6.60	12.63	13.00	83.78
LightMem [Fang et al., 2025a]	43.40	14.45	2.28	15.14	6.69	13.59	13.85	84.01
MemoType (Ours)	**56.80**	**20.13**	**4.66**	**21.83**	**10.50**	**20.09**	**20.18**	**85.19**
LoCoMo								
HyDE [Gao et al., 2023]	40.18	14.86	2.60	15.25	7.35	14.14	14.15	84.48
Mill [Jia et al., 2024]	37.41	14.50	2.49	14.88	7.13	13.84	13.83	84.38
Query2Doc [Wang et al., 2023a]	39.83	14.57	2.50	14.95	7.16	13.88	13.87	84.44
SeCom [Pan et al., 2025]	38.57	14.85	2.41	15.30	7.37	14.25	14.25	84.44
HippoRAG2 [Gutiérrez et al., 2025]	37.81	14.61	2.56	15.03	7.33	14.03	14.04	84.42
RAPTOR [Sarthi et al., 2024]	31.72	14.55	2.88	15.09	7.49	14.18	14.17	84.48
A-Mem [Xu et al., 2025c]	39.98	14.44	2.32	14.82	6.85	13.73	13.75	84.39
MemoryOS [Kang et al., 2025]	36.81	13.33	2.12	13.81	6.37	12.85	12.84	84.20
LightMem [Fang et al., 2025a]	25.03	9.20	1.20	9.70	3.70	8.85	8.85	83.30
MemoType (Ours)	**48.79**	**22.22**	**4.66**	**23.24**	**11.88**	**21.84**	**21.80**	**85.82**
LongMemEval-M								
HyDE [Gao et al., 2023]	34.60	9.01	1.26	9.71	3.99	8.46	8.67	82.70
Mill [Jia et al., 2024]	35.40	8.79	1.29	9.47	3.89	8.14	8.39	82.58
Query2Doc [Wang et al., 2023a]	34.80	8.66	1.20	9.34	3.88	7.99	8.23	82.61
SeCom [Pan et al., 2025]	34.40	10.13	1.37	10.83	4.65	9.51	9.69	83.07
MemoryOS [Kang et al., 2025]	36.20	12.22	1.79	12.87	5.87	11.49	11.79	83.57
LightMem [Fang et al., 2025a]	25.20	9.30	1.11	9.92	4.15	8.65	8.92	82.43
MemoType (Ours)	**50.20**	**18.77**	**4.24**	**20.36**	**9.19**	**18.56**	**18.64**	**85.02**
PerLTQA								
HyDE [Gao et al., 2023]	50.56	33.96	8.48	35.61	19.55	29.69	29.74	89.47
Mill [Jia et al., 2024]	60.11	38.55	11.21	40.17	23.44	34.00	34.05	90.26
Query2Doc [Wang et al., 2023a]	52.85	35.01	9.14	36.64	20.42	30.65	30.74	89.66
HippoRAG2 [Gutiérrez et al., 2025]	56.73	37.04	10.11	38.71	22.17	32.75	32.86	89.97
RAPTOR [Sarthi et al., 2024]	40.84	32.51	8.56	34.38	18.3	28.65	28.72	89.13
A-Mem [Xu et al., 2025c]	52.01	35.13	9.24	36.76	20.58	30.86	30.93	89.62
SeCom [Pan et al., 2025]	57.68	37.49	10.27	39.18	21.91	32.85	32.92	90.09
MemoryOS [Kang et al., 2025]	60.08	36.32	9.89	37.95	21.46	31.73	31.79	89.93
LightMem [Fang et al., 2025a]	31.36	32.13	7.69	34.12	16.66	28.01	28.06	89.15
MemoType (Ours)	**62.83**	**47.26**	**20.65**	**51.20**	**33.02**	**44.78**	**44.79**	**91.61**

Table 3: The ablation study of the proposed type-aware strategy with single retrieval strategy on three datasets.

Strategy	LongMemEval-S			LoCoMo			PerLTQA		
	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5
Key Expansion	51.77	76.62	85.59	25.93	43.25	51.06	66.10	83.64	88.05
Hypothetical Query	53.91	78.29	87.06	24.42	43.00	50.10	62.65	82.90	89.08
Hypothetical Memory	52.03	77.45	86.22	25.23	42.30	51.11	66.24	83.88	88.34
Type-Aware Strategy	**59.15**	**83.19**	**90.21**	**31.07**	**47.73**	**54.08**	**71.92**	**88.05**	**92.78**
Improvement	5.24	4.90	3.15	5.14	4.48	2.97	5.68	4.17	3.70

across all four datasets. On LongMemEval-S, MemoType attains a GPT4Judge score of 56.80 and a BLEU score of 4.66, surpassing the strongest baseline LightMem by over 13 points and 2.38 points respectively, which highlights the limitations of unified retrieval strategies in generating high-quality answers. On LoCoMo, MemoType outperforms all baselines by a notable margin, achieving an F1 of 22.22 and a GPT4Judge of 48.79, confirming its effectiveness in leveraging memory diversity for user-user dialogue scenarios. These results underscore the robustness and generalizability of our method across diverse memory benchmarks.

4.3 Ablation Study

In this section, we conduct ablation studies to analyze the effectiveness of our proposed method. Due to space limits, more experiments about router performance, cost analysis, retrieval strategy ablation, and other ablation studies can be found in Appendix D.

Type-Aware Retrieval Strategy. To validate the type-aware retrieval strategy, we compare it with three single-strategy baselines where all memory types employ the same retrieval enhancement, as shown in Table 3. Our type-aware strategy consistently outperforms all single strategies across three datasets. On LongMemEval-S, our strategy achieves a Recall@1 of 59.15, surpassing the best single strategy Hypothetical Query by 5.24 points. Similar improvements are observed on LoCoMo and PerLTQA, with gains of 5.14 and 5.68 in Recall@1 respectively. These results confirm that assigning tailored retrieval strategies to different memory types effectively leverages their unique characteristics and yields substantial performance gains over any uniform approach.

Memory Pruning Module. To evaluate the effectiveness of the proposed query-conditioned memory pruning module, we compare the generation performance with and without pruning, as shown in Figure 3. On LongMemEval-S, applying the pruning module improves F1 from 19.55 to 20.31. These improvements demonstrate that retrieved memory segments inevitably contain query-irrelevant information that introduces noise into the generation process. By adaptively filtering out such irrelevant content before feeding memories into the LLM, the pruning module effectively reduces noise and enhances the quality of generated responses across different datasets. Due to the space limits, more experimental results about pruning module ablation can be found in Appendix D.3.

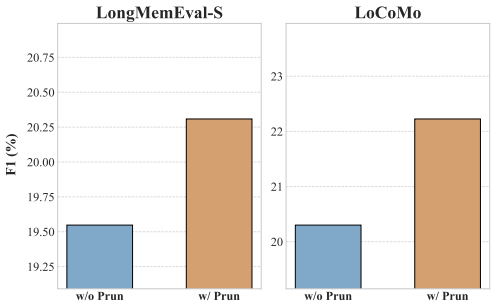


Figure 3: The ablation study of the memory pruning module on two benchmarks.

5 Related Work

RAG-based Memory technology RAG-based memory strategies can be broadly classified into unstructured and structured types. Unstructured RAG [Lu et al., 2023, Zhang et al., 2025a, Yang et al., 2025b, Zhong et al., 2024, Chhikara et al., 2025] stores history as dense chunks or LLM summaries and retrieves by similarity; representative systems include SeCom's segment-level denoising [Pan et al., 2025]. Structured RAG [Gutiérrez et al., 2024, Edge et al., 2024, He et al., 2024, Jin et al., 2024, Guo et al., 2024] introduces relations via graphs or trees: Zep's [Rasmussen et al., 2025] Graphiti integrates conversations and business data into a temporal KG, MemTree [Rezazadeh et al., 2024] organizes a dynamic tree, and H-MEM [Sun and Zeng, 2025]. Structural information augmentation can improve performance on complex reasoning tasks, but it often incurs inevitable computational overhead. For simpler problems, such enhancements may introduce noise and degrade performance. Our proposed Memotype addresses this issue effectively by adopting tailored strategies for each.

Agent Memory Beyond RAG, agents learn or structure memory with richer mechanisms [Wang et al., 2024, Wang et al., Wang and Chen, 2025, Kang et al., 2025, Hu et al., 2025]. MemoryLLM [Wang et al., 2024] updates parameters with a latent memory pool. Memory-R1 [Yan et al., 2025] trains a memory manager agent with RL to add/update/delete entries. Learn-to-Memorize models memory cycles with MoE-gated retrieval, aggregation, and reflection; Mem^p [Fang et al., 2025b] distills procedural traces into lifelong skills. Through methods such as reinforcement learning, these approaches often achieve superior performance compared to traditional RAG systems. However, they typically incur higher memory construction latency. Balancing performance and latency remains an area worthy of further exploration for these methodologies.

6 Conclusion

In this paper, we propose **TriMEM**, a multi-class memory benchmark inspired by cognitive psychology, categorizing memory into *Episodic Memory*, *Personal Semantic Memory*, and *General Semantic Memory*. Built on this foundation, we introduce **MemoType**, a novel memory framework that adaptively identifies memory types and designs tailored storage and retrieval strategies, significantly enhancing performance. We further provide a theoretical analysis showing that single-strategy retrieval faces fundamental precision bounds in heterogeneous corpora. Extensive experiments demonstrate its effectiveness. Whether different memory types at the corpus levels can benefit from better strategies remains a promising direction for future research.

Acknowledgments and Disclosure of Funding

This research was partially supported by National Natural Science Foundation of China (No.62502404), Hong Kong Research Grants Council (Research Impact Fund No.R1015-23, Collaborative Research Fund No.C1043-24GF, RGC Research Fellow Scheme No.RFS2627-1S03, General Research Fund No.11218325, No.11212926), Institute of Digital Medicine of City University of Hong Kong (No.9229503), and Huawei (Huawei Innovation Research Program).

References

Sanghwan Bae, Donghyun Kwak, Soyoung Kang, Min Young Lee, Sungdong Kim, Yuin Jeong, Hyeri Kim, Sang-Woo Lee, Woomyoung Park, and Nako Sung. Keep me updated! memory management in long-term conversations. *arXiv preprint arXiv:2210.08750*, 2022.

Mingyang Chen, Linzhuang Sun, Tianpeng Li, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, et al. Learning to reason with search for llms via reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Nuo Chen, Hongguang Li, Jianhui Chang, Juhua Huang, Baoyuan Wang, and Jia Li. Compress to impress: Unleashing the potential of compressive memory in real-world long-term conversations. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 755–773, 2025.

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

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. Perltqa: A personal long-term memory dataset for memory classification, retrieval, and fusion in question answering. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*, pages 152–164, 2024.

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

Michael W Eysenck and Mark T Keane. *Cognitive psychology: A student's handbook*. Psychology press, 2020.

Jizhan Fang, Xinle Deng, Haoming Xu, Ziyan Jiang, Yuqi Tang, Ziwen Xu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, et al. Lightmem: Lightweight and efficient memory-augmented generation. *arXiv preprint arXiv:2510.18866*, 2025a.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. Memp: Exploring agent procedural memory. *arXiv preprint arXiv:2508.06433*, 2025b.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777, 2023.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. Lightrag: Simple and fast retrieval-augmented generation. 2024.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: neurobiologically inspired long-term memory for large language models. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pages 59532–59569, 2024.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From rag to memory: Non-parametric continual learning for large language models. *arXiv preprint arXiv:2502.14802*, 2025.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37:132876–132907, 2024.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, et al. Memory in the age of ai agents. *arXiv preprint arXiv:2512.13564*, 2025.

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. *arXiv preprint arXiv:2112.09118*, 2021.

Pengyue Jia, Yiding Liu, Xiangyu Zhao, Xiaopeng Li, Changying Hao, Shuaiqiang Wang, and Dawei Yin. Mill: Mutual verification with large language models for zero-shot query expansion. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2498–2518, 2024.

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

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, et al. Personamem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025b.

Yi Jiang, Sendong Zhao, Jianbo Li, Haochun Wang, and Bing Qin. Gainrag: Preference alignment in retrieval-augmented generation through gain signal synthesis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10746–10757, 2025c.

Yi Jiang, Sendong Zhao, Jianbo Li, Haochun Wang, Lizhe Zhang, Yan Liu, and Bing Qin. Collaborative chain-of-agents for parametric-retrieved knowledge synergy. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3663–3680, 2026.

Bowen Jin, Chulin Xie, Jiawei Zhang, Kashob Kumar Roy, Yu Zhang, Zheng Li, Ruirui Li, Xianfeng Tang, Suhang Wang, Yu Meng, et al. Graph chain-of-thought: Augmenting large language models by reasoning on graphs. *arXiv preprint arXiv:2404.07103*, 2024.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

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*, pages 25972–25981, 2025.

Seo Hyun Kim, Tzu-iunn Ong, Taeyoon Kwon, Namyoung Kim, Keummin Ka, SeongHyeon Bae, Yohan Jo, Seung-won Hwang, Dongha Lee, Jinyoung Yeo, et al. Theanine: Revisiting memory management in long-term conversations with timeline-augmented response generation. *arXiv e-prints*, pages arXiv–2406, 2024.

Xiaopeng Li, Pengyue Jia, Derong Xu, Yi Wen, Yingyi Zhang, Wenlin Zhang, Wanyu Wang, Yichao Wang, Zhaocheng Du, Xiangyang Li, et al. A survey of personalization: From rag to agent. *ACM Transactions on Information Systems*, 44(4):1–39, 2026.

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

Lei Liu, Xiaoyan Yang, Yue Shen, Binbin Hu, Zhiqiang Zhang, Jinjie Gu, and Guannan Zhang. Think-in-memory: Recalling and post-thinking enable llms with long-term memory. *arXiv preprint arXiv:2311.08719*, 2023.

Shuochen Liu, Junyi Zhu, Long Shu, Junda Lin, Yuhao Chen, Haotian Zhang, Chao Zhang, Derong Xu, Jia Li, Bo Tang, et al. Perma: Benchmarking personalized memory agents via event-driven preference and realistic task environments. *arXiv preprint arXiv:2603.23231*, 2026.

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

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

Jiayan Nan, Wenquan Ma, Wenlong Wu, and Yize Chen. Nemori: Self-organizing agent memory inspired by cognitive science. *arXiv preprint arXiv:2508.03341*, 2025.

Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seung-won Hwang, Dongha Lee, and Jinyoung Yeo. Towards lifelong dialogue agents via timeline-based memory management. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8631–8661, 2025.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Xufang Luo, Hao Cheng, Dongsheng Li, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Jianfeng Gao. Secom: On memory construction and retrieval for personalized conversational agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xKDZAW0He3.

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*, pages 1–22, 2023.

Zackary Rackauckas. Rag-fusion: a new take on retrieval-augmented generation. *arXiv preprint arXiv:2402.03367*, 2024.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: a temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*, 2025.

Alireza Rezazadeh, Zichao Li, Wei Wei, and Yujia Bao. From isolated conversations to hierarchical schemas: Dynamic tree memory representation for llms. *arXiv preprint arXiv:2410.14052*, 2024.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GN921JHCRw.

Yaorui Shi, Yuxin Chen, Siyuan Wang, Sihang Li, Hengxing Cai, Qi Gu, Xiang Wang, and An Zhang. Look back to reason forward: Revisitable memory for long-context llm agents. *arXiv preprint arXiv:2509.23040*, 2025.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33: 16857–16867, 2020.

Haoran Sun and Shaoning Zeng. Hierarchical memory for high-efficiency long-term reasoning in llm agents. *arXiv preprint arXiv:2507.22925*, 2025.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, et al. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8416–8439, 2025.

Endel Tulving et al. Episodic and semantic memory. *Organization of memory*, 1(381-403):1, 1972.

Xinhang Wan, Jiyuan Liu, Xinbiao Gan, Xinwang Liu, Siwei Wang, Yi Wen, Tianjiao Wan, and En Zhu. One-step multi-view clustering with diverse representation. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3):5774–5786, 2024a.

Xinhang Wan, Bin Xiao, Xinwang Liu, Jiyuan Liu, Weixuan Liang, and En Zhu. Fast continual multi-view clustering with incomplete views. *IEEE Transactions on Image Processing*, 33:2995–3008, 2024b.

Liang Wang, Nan Yang, and Furu Wei. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423, 2023a.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. Augmenting language models with long-term memory. *Advances in Neural Information Processing Systems*, 36:74530–74543, 2023b.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788, 2020.

Yu Wang and Xi Chen. Mirix: Multi-agent memory system for llm-based agents. *arXiv preprint arXiv:2507.07957*, 2025.

Yu Wang, Dmitry Krotov, Yuanzhe Hu, Yifan Gao, Wangchunshu Zhou, Julian McAuley, Dan Gutfreund, Rogerio Feris, and Zexue He. M+: Extending memoryllm with scalable long-term memory. In *Forty-second International Conference on Machine Learning*.

Yu Wang, Yifan Gao, Xiusi Chen, Haoming Jiang, Shiyang Li, Jingfeng Yang, Qingyu Yin, Zheng Li, Xian Li, Bing Yin, et al. Memoryllm: Towards self-updatable large language models. In *International Conference on Machine Learning*, pages 50453–50466. PMLR, 2024.

Yu Wang, Ryuichi Takanobu, Zhiqi Liang, Yuzhen Mao, Yuanzhe Hu, Julian McAuley, and Xiaojian Wu. Mem-{\alpha}: Learning memory construction via reinforcement learning. *arXiv preprint arXiv:2509.25911*, 2025.

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

Derong Xu, Xinhang Li, Ziheng Zhang, Zhenxi Lin, Zhihong Zhu, Zhi Zheng, Xian Wu, Xiangyu Zhao, Tong Xu, and Enhong Chen. Harnessing large language models for knowledge graph question answering via adaptive multi-aspect retrieval-augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25570–25578, 2025a.

Derong Xu, Yi Wen, Pengyue Jia, Yingyi Zhang, Yichao Wang, Huifeng Guo, Ruiming Tang, Xiangyu Zhao, Enhong Chen, Tong Xu, et al. From single to multi-granularity: Toward long-term memory association and selection of conversational agents. *arXiv preprint arXiv:2505.19549*, 2025b.

Derong Xu, Shuochen Liu, Pengfei Luo, Pengyue Jia, Yingyi Zhang, Yi Wen, Yimin Deng, Wenlin Zhang, Enhong Chen, Xiangyu Zhao, et al. Learning how and what to memorize: Cognition-inspired two-stage optimization for evolving memory. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 44987–45011, 2026.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*, 2023.

Wujiang Xu, Kai Mei, Hang Gao, Juntao Tan, Zujie Liang, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*, 2025c.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Hinrich Schütze, Volker Tresp, and Yunpu Ma. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*, 2025.

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

Wei Yang, Jinwei Xiao, Hongming Zhang, Qingyang Zhang, Yanna Wang, and Bo Xu. Coarse-to-fine grounded memory for llm agent planning. *arXiv preprint arXiv:2508.15305*, 2025b.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, et al. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*, 2025.

Shengju Yu, Suyuan Liu, Siwei Wang, Chang Tang, Zhigang Luo, Xinwang Liu, and En Zhu. Sparse low-rank multi-view subspace clustering with consensus anchors and unified bipartite graph. *IEEE Transactions on Neural Networks and Learning Systems*, 36(1):1438–1452, 2023a.

Shengju Yu, Siwei Wang, Yi Wen, Ziming Wang, Zhigang Luo, En Zhu, and Xinwang Liu. How to construct corresponding anchors for incomplete multiview clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(4):2845–2860, 2023b.

Yi Yu, Liuyi Yao, Yuexiang Xie, Qingquan Tan, Jiaqi Feng, Yaliang Li, and Libing Wu. Agentic memory: Learning unified long-term and short-term memory management for large language model agents. *arXiv preprint arXiv:2601.01885*, 2026.

Gaoke Zhang, Bo Wang, Yunlong Ma, Dongming Zhao, and Zifei Yu. Multiple memory systems for enhancing the long-term memory of agent. *arXiv preprint arXiv:2508.15294*, 2025a.

Sheng Zhang, Junyi Li, Yingyi Zhang, Pengyue Jia, Yichao Wang, Xiaowei Qian, Wenlin Zhang, Maolin Wang, Yong Liu, and Xiangyu Zhao. Memsearch-o1: Empowering large language models with reasoning-aligned memory growth in agentic search. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 925–943, 2026a.

Wenlin Zhang, Xiaopeng Li, Yingyi Zhang, Pengyue Jia, Yichao Wang, Huifeng Guo, Yong Liu, and Xiangyu Zhao. Deep research: A survey of autonomous research agents. *arXiv preprint arXiv:2508.12752*, 2025b.

Wenlin Zhang, Xiangyang Li, Kuicai Dong, Yichao Wang, Pengyue Jia, Xiaopeng Li, Yingyi Zhang, Derong Xu, Zhaocheng Du, Huifeng Guo, et al. Process vs. outcome reward: Which is better for agentic rag reinforcement learning. *Advances in Neural Information Processing Systems*, 38: 58701–58729, 2026b.

Yingyi Zhang, Junyi Li, Wenlin Zhang, Pengyue Jia, Xianneng Li, Yichao Wang, Derong Xu, Yi Wen, Huifeng Guo, Yong Liu, et al. Evoking user memory: Personalizing llm via recollection-familiarity adaptive retrieval. In *International Conference on Learning Representations*, volume 2026, pages 54178–54213, 2026c.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao.

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

Technical Appendix

Table of Contents

A Proof of Theorem 3.3 **17**

B TriMEM Dataset Details **17**

B.1 Classification Rationale 17

B.2 Memory Type Definitions 18

B.3 Dataset Construction Pipeline 19

B.4 Dataset Statistics 20

B.5 Router Classification Performance 21

B.6 Data Leakage Analysis 21

B.7 Dataset Statistics 22

C Benchmark and Baseline Details **22**

C.1 Evaluation Dataset Details 22

C.2 Baseline Descriptions 24

D Additional Experiments **25**

D.1 Retriever Ablation 25

D.2 Router Performance Ablation 26

D.3 Memory Pruning Ablation 27

D.4 Per-Type Retrieval Strategy Contributions 27

D.5 Router Architecture Ablation 28

D.6 Cost Analysis 28

D.7 Empirical Evidence for Semantic Clustering 29

E Extended Related Work **30**

E.1 RAG-based Memory Methods 30

E.2 RL-based Memory Methods 31

F Prompts **31**

A Proof of Theorem 3.3

Theorem 3.3 (Upper Bound for Single Retrieval)**.** *Let $\mathcal{D}$ satisfy Assumption 3.1 with angular separation $\delta > 0$. Assume documents are uniformly distributed across C categories. Then, for any retrieval encoder ϕ, the expected query-document similarity has an upper bound:*

$$\mathbb{E}_{q\sim\mathcal{Q},d\sim\mathcal{D}}[\langle e_{query}, e_{doc}\rangle] \leq 1 - \frac{(C-1)\delta}{2C},$$

where $e_{query} = \phi(q)/\|\phi(q)\|$ and $e_{doc} = \phi(d)/\|\phi(d)\|$.

Proof. Let $u = \mathbb{E}_{q\sim\mathcal{Q}}[e_{query}]$ and $\mu_i = \mathbb{E}_{d\in S_i}[e_{doc}]$ for each category S_i. Since $\phi(\cdot)$ are ℓ_2-normalized, we have $\mathbb{E}_{d\in S_i}[e_{doc}] = \mu_i = \|\mu_i\|v_i^*$ for each category, where $\|\mu_i\| \leq 1$ by Jensen's inequality. Besides, with queries and documents sampled independently, we have:

$$\mathbb{E}_{q\sim\mathcal{Q},d\sim\mathcal{D}}[\langle e_{query}, e_{doc}\rangle] = \langle \mathbb{E}_{q\sim\mathcal{Q}}[e_{query}], \mathbb{E}_{d\sim\mathcal{D}}[e_{doc}]\rangle = \langle u, \mathbb{E}_{d\sim\mathcal{D}}[e_{doc}]\rangle.$$

Since documents are uniformly distributed across C categories:

$$\mathbb{E}_{d\sim\mathcal{D}}[e_{doc}] = \frac{1}{C}\sum_{i=1}^{C}\mathbb{E}_{d_i\sim\mathcal{D}_i}[e_d^i] = \frac{1}{C}\sum_{i=1}^{C}\|\mu_i\|v_i^*.$$

Thus:

$$\mathbb{E}_{q\sim\mathcal{Q},d\sim\mathcal{D}}[\langle e_{query}, e_{doc}\rangle] = \frac{1}{C}\sum_{i=1}^{C}\|\mu_i\|\langle u, v_i^*\rangle = \langle u, \bar{v}\rangle,$$

where $\bar{v} = \frac{1}{C}\sum_{i=1}^{C}\|\mu_i\|v_i^*$. By Cauchy-Schwarz inequality and $\|u\| \leq 1$:

$$\langle u, \bar{v}\rangle \leq \|u\|\|\bar{v}\| \leq \|\bar{v}\|.$$

We now bound $\|\bar{v}\|^2$:

$$\|\bar{v}\|^2 = \frac{1}{C^2}\sum_{i,j}\|\mu_i\|\|\mu_j\|\langle v_i^*, v_j^*\rangle = \frac{1}{C^2}\left[\sum_i\|\mu_i\|^2 + \sum_{i\neq j}\|\mu_i\|\|\mu_j\|\langle v_i^*, v_j^*\rangle\right].$$

Since $\|\mu_i\| \leq 1$ for all i. By Assumption 3.1, $\langle v_i^*, v_j^*\rangle \leq 1-\delta$ for $i \neq j$:

$$\|\bar{v}\|^2 \leq \frac{1}{C^2}[C + C(C-1)(1-\delta)] = 1 - \frac{(C-1)\delta}{C}.$$

Using $\sqrt{1-x} \leq 1 - x/2$ for $x \in [0,1]$:

$$\|\bar{v}\| \leq 1 - \frac{(C-1)\delta}{2C}.$$

Therefore,

$$\mathbb{E}_{q\sim\mathcal{Q},d\sim\mathcal{D}}[\langle e_{query}, e_{doc}\rangle] \leq \|\bar{v}\| \leq 1 - \frac{(C-1)\delta}{2C}.$$

□

Since higher alignment $\langle u, v_i^*\rangle$ leads to higher retrieval scores for category S_i documents, this bound implies a fundamental limit on retrieval performance: no single retrieval direction can simultaneously achieve high alignment with all category directions.

B TriMEM Dataset Details

B.1 Classification Rationale

In this paper, we will focus on text classification in the form of dialogue. Because dialogues are crucial retrieval objects in existing memory scenarios. For instance, in the two common memory benchmarks

Table 4: The Memory Classification Criterion in the TriMEM Benchmark. "EM" denotes Episodic Memory, "PM" denotes the Personal Semantic Memory, "GM" denotes the General Semantic Memory.

Memory Type	Content Type	Example
EM	Event Occurred	"Sarah and John had dinner at a sushi restaurant downtown."
	Events Planned to Occur	"Next Month, my colleagues are planning to see a beautiful sunset."
PM	User Information	"Jack's degree is in Mathematics. "
	User Preferences and Habits	"I love Italian food, especially pasta and pizza."
	User Intent	"I want to buy a pair of running shoes"
GM	General Knowledge	"Water boils at 100 degrees Celsius at sea level."
	Broadly Applicable Suggestions	"Getting up early is a good habit for maintaining good health."
	Objective Descriptions of Entities	"The desks in the school are made of wood."
None	Question without Information	"Do you have any tips on how to organize a shoe rack?"
	Sentence without Information	"Good Morning, Peter"; "See you next time"

LoCoMo [Maharana et al., 2024] and LongMemEval [Wu et al., 2025], dialogues are the primary and exclusive retrieval objects. Moreover, in real-world scenarios, accessible data typically exists in dialogue form. We propose three memory types: Episodic Memory, Personal Semantic Memory, and General Semantic Memory. This classification is rooted in cognitive psychology research on declarative memory [Eysenck and Keane, 2020], which encompasses memory for facts and events that can be consciously recalled and linguistically described. In this subsection, we will explain the basis for our classification from the perspectives of completeness and necessity. Specifically, we will demonstrate that the three categories of memory encompass all conversational memory and justify the necessity of this classification.

Completeness. Our classification is based on well-established research in cognitive psychology [Eysenck and Keane, 2020] on declarative memory. Declarative memory inherently includes both semantic memory and episodic memory, which collectively cover all forms of dialogue memory that can be expressed through language. Nondeclarative memory, such as procedural memory or priming memory, cannot be articulated linguistically and thus lies beyond the scope of this framework. Therefore, the three proposed categories comprehensively encompass all dialogue memory types without requiring additional classifications.

Necessity. The necessity of categorizing semantic memory into Personal Semantic Memory and General Semantic Memory is grounded in two critical considerations. First, the division addresses the category imbalance problem, as semantic memory accounts for a disproportionately large proportion (87.64%) of dialogue memory compared to episodic memory (12.35%). Subdividing semantic memory ensures a more balanced classification, mitigating the dominance of one category during model optimization and evaluation. Second, the semantic gap between Personal and General Semantic Memory is empirically validated through TSNE visualization and word cloud analysis. As shown in the Figure 1a, both categories exhibit distinct clustering patterns, indicating significant semantic differences between personal and general semantic memory. Additionally, we supplemented PM Word Cloud Visualizations and GM Word Cloud Visualizations as shown in Figure 5 and Figure 6, revealing that General Semantic Memory emphasizes common terms like "assistant" and "offer," while Personal Semantic Memory focuses on user-specific terminology such as "planning "and "user." This further highlights the distinctions between the two memory types. This clear semantic distinction underscores the necessity of separating semantic memory into these two subcategories.

B.2 Memory Type Definitions

In this subsection, we offer a more comprehensive explanation of the three types of memory, as shown in Table 4.

- **Episodic Memory.** Episodic memory refers to memories of specific events or occurrences tied to contextual details such as time, location, participants, and activities. These memories can include both past experiences and plans for the future. For example: "Last summer, I attended a three-day conference in Paris where I presented my research on renewable energy." "Next Friday, I'm scheduled to meet my supervisor at 10 a.m. to review the final draft of my thesis."

- **Personal Semantic Memory.** Personal Semantic Memory refers to long-term knowledge related to an individual's identity, preferences, habits, or intentions. Unlike Episodic Memory, it focuses on general personal information rather than specific events. For instance: "I have been working as a data scientist for a long time, and I specialize in natural language processing." "I usually prefer hiking in the mountains over beach vacations because I enjoy the tranquility of nature."
- **General Semantic Memory.** General Semantic Memory encompasses universally applicable knowledge, objective descriptions, or facts about the world. It is not tied to any specific individual or event. Examples include: "The Great Wall of China is over 13,000 miles long and was constructed to protect against invasions." "Photosynthesis is the process by which green plants convert sunlight into chemical energy stored in glucose."

B.3 Dataset Construction Pipeline

TriMEM is a carefully curated dataset comprising 6,000 dialogue samples, designed to evaluate long-term memory capabilities in dialogue models. The dataset is constructed with diversity and quality in mind, and its creation involves multiple steps including sampling, rewriting, and labeling, all of which are detailed below.

Data Sampling. TriMEM is composed of dialogue samples sourced from multiple datasets and manually constructed dialogues to ensure a balanced and realistic testbed. *LongMemEval-S*: 1,000 dialogue pairs were sampled from this dataset, which contains approximately 20,000 dialogue records. To prevent over-representation, the sampling proportion was kept low at only 10%. *LoCoMo*: 750 dialogue pairs were sampled to add variety. *PersonaMem*: 750 dialogue pairs were also sampled to further diversify the dataset. *Manually-constructed dialogues*: 500 dialogue pairs were created by human annotators to introduce additional variability and ensure coverage of edge cases not captured in the sampled datasets.

Benchmark Selection. We aimed to annotate memory categories in real-world dialogue scenarios rather than generating memory-related dialogues from predefined categories, which would risk introducing bias by over-focusing on single categories. We chose LoCoMo [Maharana et al., 2024] and LongMemEval [Wu et al., 2025] as they are widely recognized memory datasets frequently referenced in works such as A-Mem [Xu et al., 2025c], mem0 [Chhikara et al., 2025], MIRIX [Wang and Chen, 2025], and Secom [Pan et al., 2025]. This highlights their significance and credibility. LongMemEval emphasizes user-assistant interactions, while LoCoMo focuses on user-user dialogues, aligning with our goal of covering diverse memory scenarios and topics. To further enhance the generalizability of our annotated dataset beyond evaluation-specific categories, we included PersonaMem [Jiang et al., 2025a], a personalized memory benchmark that introduces varied user profiles. This addition strengthens the dataset's applicability across broader dialogue classification tasks.

Rewriting to Prevent Overfitting. Since both LongMemEval-S and LoCoMo are also used in the evaluation datasets of this study, special care was taken to avoid the risk of the model directly memorizing specific dialogue samples. Such memorization could compromise the evaluation of classification performance.

To address this, the sampled dialogues from these datasets were rewritten using transfer prompt, which is shown in Figure 14 and Figuer 15. The rewriting process involved modifying the necessary entity names, rephrasing relevant topic statements, and substituting key details while preserving the original semantic structure. This ensured that the dialogues remained contextually meaningful but sufficiently distinct from their original forms, mitigating the risks of data leakage and overfitting.

Labeling Process. The labeling of dialogue samples was conducted through a combination of automated and manual processes to ensure high accuracy and consistency. The GPT-5.2-2025-12-11 model was used to generate initial labels for the dialogue samples. These prompts were meticulously crafted and validated on a subset of the data to ensure their effectiveness for the classification task. Notably, we opted for separate prompts for each category, as this approach achieved higher classification accuracy compared to using specific prompts for all categories (Episodic Memory Classification Prompt in Figure 16, Personal Semantic Memory Classification Prompt in Figure 17, and General Semantic Memory Classification Prompt in Figure 18). These prompts were tailored to the unique characteristics of each memory category, ensuring precise and context-aware labeling.

Human Verification To ensure the quality and reliability of the TriMEM dataset, we conducted a rigorous human verification process. Three PhD-level annotators with expertise in natural language

processing and dialogue systems were recruited to independently review the dataset annotations. *(i) Verification Protocol.* A stratified random sample of 1,200 dialogue pairs (20% of the total dataset) was selected for manual review, covering all source datasets proportionally: 400 samples from LongMemEval-S, 300 from LoCoMo, 300 from PersonaMem, and 200 from manually-constructed dialogues. Each sample was independently evaluated by at least two annotators, and disagreements were resolved through discussion until consensus was reached. *(ii) Evaluation Criteria.* Annotators assessed each sample along three dimensions: 1) Category Correctness: whether the assigned memory category accurately reflects the dialogue content; 2) Rewriting Quality: whether the rewritten dialogues preserve the original semantic structure while being sufficiently distinct from the source; and 3) Label Consistency: whether the annotations are internally consistent and free of contradictions. *(iii) Results.* The inter-annotator agreement, measured by Fleiss' κ, was 0.87, indicating near-perfect agreement. The overall annotation accuracy was 94.6%, with category correctness at 95.2%, rewriting quality at 93.8%, and label consistency at 94.9%. For the 64 samples (5.4%) identified as erroneous or ambiguous, annotators provided corrected labels, which were incorporated into the final dataset. A second round of verification on the corrected samples confirmed 100% agreement among all three annotators. *iv: Quality Assurance.* Beyond the sampled verification, annotators also flagged 23 additional edge cases during the review process that required further deliberation. These cases were discussed in a joint session and resolved with detailed annotation guidelines, which were subsequently applied to re-examine all similar instances in the full dataset. This iterative process further strengthened the overall quality and consistency of TriMEM.

B.4 Dataset Statistics

The **TriMEM** dataset comprises 6,000 dialogues. User-assistant interactions account for 58.77% of the data and user-user interactions for 41.23yielding a roughly 60/40 split. Each text contains an average of 91.06 tokens, indicating moderate linguistic complexity.

Coverage across the four memory categories is well distributed. Episodic Memory covers 29.08% of the dataset, Personal Semantic Memory 49.35%, General Semantic Memory 33.95%, and None 16.25%. Since a single dialogue may carry multiple labels, these proportions sum above 100%.

The breakdown by interaction type reveals clear patterns. General Semantic Memory is dominated by user-assistant dialogues at 90.48%, reflecting the assistant's role in providing factual and world knowledge. In contrast, Episodic Memory and Personal Semantic Memory slightly favor user-user dialogues, with respective shares of 59.71% and 57.72%, consistent with the more event-oriented and personal nature of peer exchanges. The None class is approximately balanced, with user-assistant turns at 56.82% and user-user turns at 43.18%.

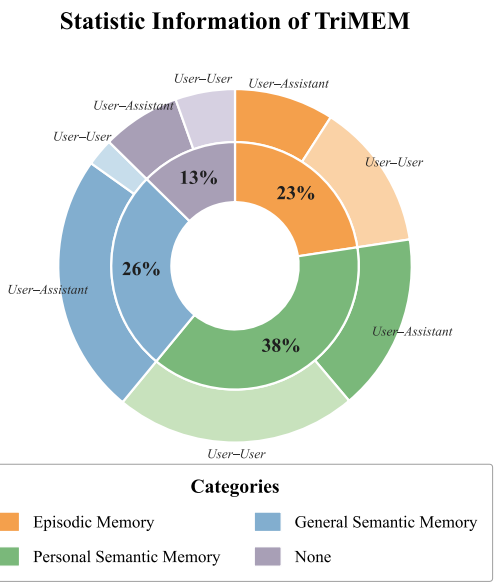


Figure 4: The Statistical Information of TriMEM[4].

This composition reflects the diversity of TriMEM and its capacity to model distinct memory types across varied conversational settings. By covering both assistant-guided and peer-to-peer exchanges, TriMEM provides a comprehensive resource for memory-aware dialogue research and supports tasks such as memory reasoning and personalized conversation.

[4]The percentage of the memory types is a relative ratio, as there is overlap between different memory types.

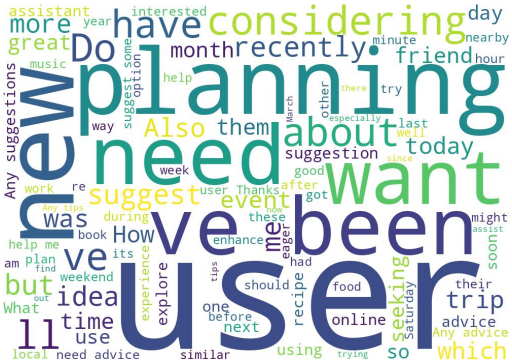


Figure 5: The Word Cloud Figure of Personal Semantic Memory on LongMemEval-S Dataset.

Figure 6: The Word Cloud Figure of General Semantic Memory on LongMemEval-S Dataset.

B.5 Router Classification Performance

In this study, we evaluated the classification performance of four large language models (LLMs) on the TriMEM dataset: Qwen3-8b [Yang et al., 2025a], Gemini-2.5-Flash [Comanici et al., 2025], GPT-4o-Mini [Hurst et al., 2024], and Qwen3-32b with multi-class prompt in Figure 19. Additionally, our model was included for comparison. The evaluation metrics used were accuracy (ACC) and F1 score, which respectively measure the overall classification correctness and the balance between precision and recall. Performance was analyzed across three memory dimensions in TriMEM: Episodic Memory (EM), Personal Semantic Memory (PM), and General Semantic Memory (GM), and PerLTQA.

Experimental results indicate that our model outperformed all other LLMs in every category. For overall accuracy and F1 score, our model achieved 92.10% and 91.75%, respectively, significantly surpassing GPT-4o-Mini, the second-best model, which scored 83.55% (ACC) and 76.94% (F1). In the EM dimension, our model achieved an outstanding 98.80% (ACC) and 97.81% (F1), outperforming GPT-4o-Mini, which achieved 85.05% (ACC) and 76.10% (F1). For PM, our model reached 82.65% (ACC) and 85.64% (F1), while GPT-4o-Mini achieved slightly higher accuracy (87.18%) but lower F1 (85.59%). Finally, in GM, our model achieved 94.85% (ACC) and 92.39% (F1), outperforming others by a clear margin.

Despite our model's superior performance, the results highlight that existing LLMs struggle to achieve precise classification on TriMEM, requiring further advancements in memory modeling for improved accuracy and generalization.

B.6 Data Leakage Analysis

As shown in Fig. 8, we compare the similarity between the transferred memories produced by **MemoType** and the original memories from the dataset using three surface-form metrics (Word Overlap Rate, BLEU, and 3-gram Overlap Rate) on LongMemEval-S and LoCoMo. Across both datasets, the distributions of all three metrics are concentrated in the low-similarity regime, and most samples fall below the threshold of 0.5. The corresponding frequency distributions in Fig. 9 and Fig. 10 further confirm this trend: the mean and median values remain low for each metric, with only a small fraction of cases exhibiting higher overlap.

To further address concerns about *semantic-level* leakage that surface-form metrics cannot capture, we additionally measure semantic similar-

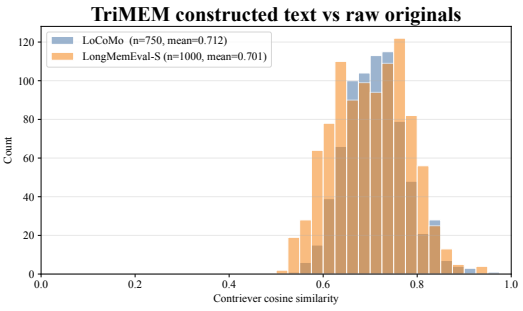


Figure 7: The Contriever Similarity Distribution between TriMem Context with the raw context on two benchmarks.

Table 5: The classification Performance on TriMEM with four LLM models. "EM" denotes Episodic Memory, "PM" denotes the Personal Semantic Memory, "GM" denotes the General Semantic Memory.

Type	Model	TriMEM(In Distribution)				PerLTQA(Out of Distribution)		
		Overall	EM	PM	GM	Profile	Relationship	Events
ACC (100 %)								
Prompting	Qwen3-8b	71.90	81.65	71.37	62.68	88.65	57.21	46.23
Prompting	Gemini-2.5-Flash	77.84	82.47	76.72	74.33	100.00	75.06	91.83
Prompting	GPT-4o-Mini	83.55	85.05	**87.18**	78.42	100.00	89.47	92.00
Prompting	Qwen3-32b	77.75	87.82	80.88	64.55	100.00	95.64	87.55
Training	Ours	**92.10**	**98.80**	82.65	**94.85**	**100.00**	**96.49**	**96.94**
F1 (100 %)								
Prompting	Qwen3-8b	55.26	49.05	55.58	61.15	93.98	72.78	63.23
Prompting	Gemini-2.5-Flash	72.75	73.47	74.55	70.22	100.00	85.75	95.74
Prompting	GPT-4o-Mini	76.94	76.10	85.59	69.14	100.00	94.44	95.83
Prompting	Qwen3-32b	72.03	76.80	75.67	63.61	100.00	98.29	93.36
Training	Ours	**91.75**	**97.81**	**85.64**	**92.39**	**100.00**	**98.45**	**98.21**

ity using Contriever embeddings, which is the same encoder employed in our retrieval experiments. As shown in Fig. 7, the Contriever cosine similarity between transferred and original memories is centered around 0.70 on both LoCoMo (mean $= 0.712$) and LongMemEval-S (mean $= 0.701$), with the distribution concentrated in the 0.6–0.8 range and only a negligible fraction exceeding 0.9. This level of semantic similarity reflects the *type-level* consistency that is required by design—rewritten dialogues must preserve enough meaning to retain their original memory-type label—rather than instance-level duplication. Notably, the absence of a high-similarity peak (> 0.9) indicates that the transferred texts are not semantic near-duplicates of their sources.

Taken together, the three surface-form metrics and the Contriever-based semantic analysis provide complementary evidence from two independent perspectives: **MemoType** preserves neither lexical patterns nor instance-level semantic content of the original benchmarks. This property is important for interpreting the role of our router: rather than relying on memorization of dataset-specific expressions or templates, the router is forced to route based on the generalized characteristics of memory types reflected in the transferred content. Consequently, the routing behavior learned by it is better aligned with the intended goal of type-level generalization, instead of reproducing or matching individual dataset instances. This conclusion is further corroborated by the strong out-of-distribution performance of the router on PerLTQA (Table 5), a benchmark that was *never* used in TriMEM construction, where the router still achieves 96–100% accuracy across all three categories.

B.7 Dataset Statistics

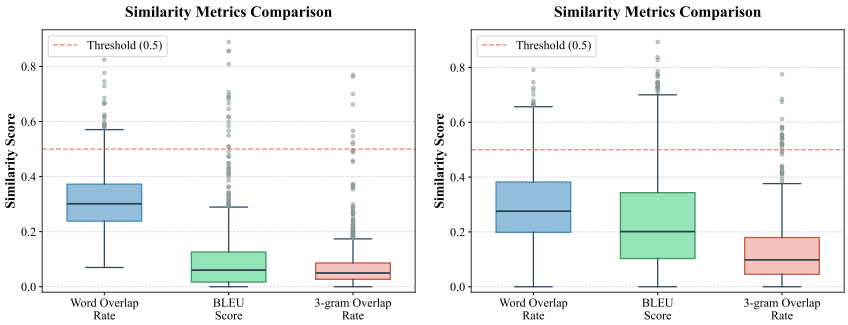
These results indicate that the transferred memories are not near-duplicates of the original texts. In other words, **MemoType** does not preserve substantial lexical or n-gram-level content from the source memories, which provides direct evidence against data leakage through copying. This property is important for interpreting the role of our router: rather than relying on memorization of dataset-specific expressions or templates, the router is forced to route based on the generalized characteristics of memory types reflected in the transferred content. Consequently, the routing behavior learned by it is better aligned with the intended goal of type-level generalization, instead of reproducing or matching individual dataset instances.

"Crucially, our router achieves 96–100% accuracy on *PerLTQA*, a benchmark whose dialogues were *never used* in TriMEM construction. This out-of-distribution performance demonstrates that the router learns generalizable type-level features rather than benchmark-specific patterns; if the router relied on dataset fingerprints, OOD performance would degrade sharply, which is not observed.

C Benchmark and Baseline Details

C.1 Evaluation Dataset Details

As statistic information in shown in Table 6, we evaluate our model with the following datasets:



(a) LongMemEval-S (b) LoCoMo

Figure 8: The Distribution of three Similarity Metrics (Word Overlap Rate, BLEU Score, and 3-gram Overlap Rate) between Transferred Memory and Original Memory on LongMemEval-S and LoCoMo datsets. The red dotted line indicates a similarity threshold of 0.5. The majority of samples fall in the low similarity region, indicating a significant difference between the transformed text and the original text.

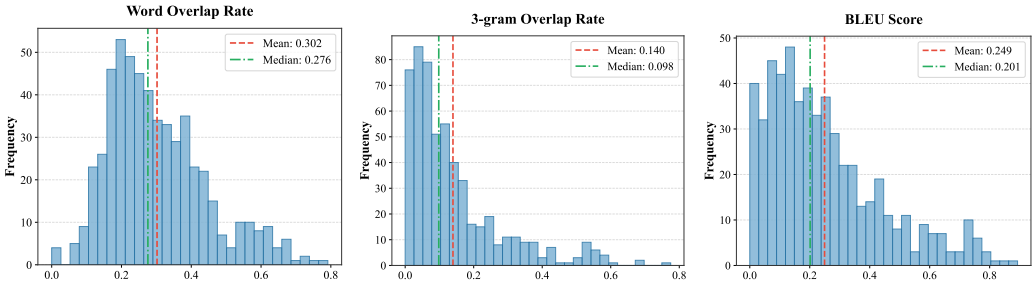


Figure 9: The Frequency Distributions of three Similarity Metrics between Transferred Memory and Original Memory on LoCoMo dataset. The red/green dashed lines indicate the mean and median, respectively. The distribution is overall low and long-tailed, further indicating that the transformed memory differs significantly from the original.

- **LongMemEval-S** [Wu et al., 2025] is a dataset designed to evaluate long-term memory in conversational AI systems. It includes 50 question sessions, each with an average of 115,000 tokens, providing a compact yet challenging benchmark. This dataset tests core memory abilities such as information extraction, multi-session reasoning, temporal reasoning, and abstention. LongMemEval-s requires chat assistants to process user-AI dialogues effectively, retain pertinent information over extended conversation histories, and maintain consistency across sessions. By focusing on shorter yet complex contexts, LongMemEval-s serves as a foundational benchmark for assessing memory performance.
- **LoCoMo** [Maharana et al., 2024] is a long-context memory benchmark, evaluates AI systems' ability to handle lengthy dialogues and long-range dependencies. It features conversations with an average of 300 turns, 9,000 tokens, and up to 35 sessions, with a publicly available subset, LoCoMo-10, comprising ten high-quality conversations. The dataset is well-suited for testing long-context language models (LLMs) and retrieval-augmented generation (RAG) systems. Despite advances, systems still struggle with temporal reasoning and maintaining coherence in extended interactions, highlighting the challenge of replicating human-like memory performance.
- **LongMemEval-M** [Wu et al., 2025] builds on LongMemEval-S with a more extensive evaluation setting, encompassing 500 sessions per question and an average of 1.5 million tokens per conversation. This dataset is designed for rigorous testing of long-term memory and scalability. It challenges systems to handle dynamic user-AI interactions across an even greater number of sessions, requiring robust memory mechanisms for historical consistency and information recall.

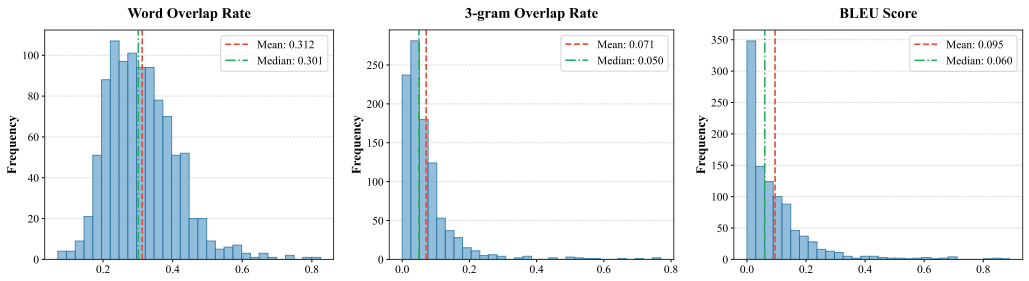


Figure 10: The Frequency Distributions of three Similarity Metrics between Transferred Memory and Original Memory on LongMemEval-S dataset. The red/green dashed lines indicate the mean and median, respectively. It is evident that all three indicators are concentrated in the lower range.

Table 6: Statistics of the long-term memory QA datasets used in our study. "#Inst." counts the top-level units. "Sess./Inst." and "Turns/Sess." are averages; "Words/Inst." is the average number of whitespace-tokenized words per instance.

Dataset	#Inst.	#QA	#Sess.	Sess./Inst.	Turns/Sess.	Words/Inst.	#Q-Types
LoCoMo	10	1,986	272	27.2	21.6	15.9 k	5
LongMemEval-S	500	500	25,097	50.2	9.8	79.0 k	6
LongMemEval-M	500	500	249,525	499.1	9.8	777.6 k	6
PerLTQA	32	2,838	128	4.0	21.7	8.9 k	4

The dataset underscores the substantial performance gap between state-of-the-art AI systems and human capabilities in long-term memory retention and reasoning over extended contexts.

- **PerLTQA** [Du et al., 2024] is a benchmark released to study personalized long-term memory in conversational AI. Each instance is built around a virtual persona and bundles four complementary memory views—structured profile, social relationships, life events, and naturalistic multi-turn dialogues—together with a large pool of human-curated question–answer pairs whose evidence is grounded in specific dialogue turns. By unifying semantic memory (facts about the user and their world) with episodic memory (what was actually said), PerLTQA enables joint evaluation of memory classification (which memory type to consult), retrieval (locating the supporting record), and fusion (synthesizing across memories) for downstream QA. Its persona-centric, multi-faceted design makes it complementary to conversation-only benchmarks such as LOCOMO and LongMemEval, and a natural testbed for memory-augmented LLM agents.

C.2 Baseline Descriptions

We evaluate our model with the following baselines:

- **Contriever** [Izacard et al., 2021] is a retrieval model that serves as a strong baseline by efficiently retrieving and ranking relevant documents based on dense representations.
- **MPNet** [Song et al., 2020] is a pre-trained transformer model that combines masked language modeling and permuted language modeling to better capture dependencies between tokens for improved natural language understanding.
- **MiniLM** [Wang et al., 2020] is a lightweight and efficient transformer-based model designed for natural language understanding tasks, offering competitive performance with significantly fewer parameters compared to larger models.
- **QAMiniLM** [Wang et al., 2020] is an extension of MiniLM, designed specifically for Question Answering tasks, leveraging MiniLM's lightweight architecture while fine-tuning it to optimize performance for extracting precise answers from text.
- **HyDE** [Gao et al., 2023] generates a "hypothetical" document using a language model and retrieves similar real documents via dense embeddings to improve zero-shot dense retrieval without needing labeled relevance data. The prompt of HyDE could be found in Figure 20.

- **Mill** [Jia et al., 2024] utilizes large language models to generate diverse sub-queries and documents, followed by a mutual verification process to synergize generated and retrieved data for effective zero-shot query expansion. The prompt of Mill could be found in Figure 21.
- **Query2Doc** [Wang et al., 2023a] improves sparse and dense retrieval systems by generating pseudo-documents through few-shot prompting of large language models (LLMs) and using them for query expansion, achieving significant performance boosts without fine-tuning. The prompt of QueryDoc could be found in Figure 22.
- **Secom** [Pan et al., 2025] enhances memory retrieval in long-term conversations by segmenting them into topically coherent units and applying compression-based denoising for improved retrieval accuracy and semantic quality.
- **HippoRAG2** [Gutiérrez et al., 2024] is a framework that enhances retrieval-augmented generation (RAG) by integrating deeper passage connections and effective LLM use, achieving superior performance on factual, sense-making, and associative memory tasks, thereby advancing non-parametric continual learning for LLMs.
- **Raptor** [Sarthi et al., 2024] improves retrieval-augmented language models by recursively embedding, clustering, and summarizing text into a hierarchical tree, enabling multi-level abstraction retrieval and achieving state-of-the-art performance in complex reasoning tasks.
- **A-Mem** [Xu et al., 2025c] introduces an agentic memory system for LLMs that dynamically organizes and evolves memories by leveraging Zettelkasten principles, enabling adaptive and context-aware knowledge management through dynamic linking and indexing.
- **MemoryOS** [Kang et al., 2025] treats LLM-agent memory as an operating-system problem, organizing it into three tiers, short-term dialogue pages, mid-term topic segments updated by a heat score (visits, length, recency), and long-term user/agent personas—and coordinating Storage, Updating, two-stage segment-then-page Retrieval, and Generation modules.
- **LightMem** [Fang et al., 2025a] follows the Atkinson–Shiffrin three-stage view: a sensory stage compresses tokens with LLMLingua-2 and segments turns by topic, a topic-aware short-term stage summarizes each chunk, and a long-term stage uses fast soft writes online plus an offline "sleep-time update" that merges and de-duplicates entries.

D Additional Experiments

D.1 Retriever Ablation

Our main experiments adopt Contriever as the default retriever. A natural question is whether the performance gains of MemoType are specific to one particular retriever or generalize across different embedding models. To investigate this, we conduct retriever ablation experiments by replacing Contriever with three alternative retrievers: MPNet, MiniLM, and QAMiniLM, covering both general-purpose and task-specific encoders.

As shown in Table 7, we evaluate MemoType against six baselines across three datasets under each retriever. Across all retriever and dataset combinations, MemoType consistently achieves the highest recall scores at ranks 1, 3, and 5. For instance, with MPNet on LongMemEval-S, MemoType achieves a Recall@1 of 47.87, outperforming the strongest baseline SeCom by over 10 points. With MiniLM on PerLTQA, MemoType reaches 49.01 in Recall@1, surpassing A-Mem by nearly 18 points. Similar trends hold for QAMiniLM, where MemoType achieves 49.79 Recall@1 on LongMemEval-S, exceeding SeCom by 8.73 points.

These results reveal two important findings. First, the type-aware retrieval strategy provides consistent improvements regardless of the underlying embedding model, confirming that the gains stem from the diversity-aware routing mechanism rather than a favorable interaction with a specific encoder. Second, the improvement margins remain substantial even with weaker retrievers, suggesting that MemoType effectively compensates for encoder limitations by directing each query to the most relevant memory subset with a tailored strategy. This demonstrates the generalizability and robustness of our framework.

Table 7: The ablation study of different retrievers.

Method	LongMemEval-S			LoCoMo			PerLTQA		
	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5
Base Retriever: MPNet									
HyDE	25.11	53.19	66.60	27.29	44.76	53.52	30.62	65.68	74.17
Mill	27.45	53.19	64.89	21.00	36.81	44.51	36.15	67.27	74.77
Query2Doc	27.45	54.68	68.72	26.38	46.12	54.03	33.51	65.82	74.63
SeCom	37.66	62.13	73.19	21.45	34.24	41.49	36.19	65.72	73.75
A-Mem	27.80	56.60	67.80	29.20	49.65	58.61	32.42	66.38	74.91
MemoryOS	31.52	57.20	65.55	23.21	42.45	51.26	42.07	65.12	72.23
Ours	**47.87**	**74.68**	**83.40**	**30.87**	**49.45**	**58.11**	**58.00**	**78.58**	**85.17**
Base Retriever: MiniLM									
HyDE	32.34	61.70	72.55	22.00	37.92	45.87	23.15	59.76	67.58
Mill	36.81	63.62	75.11	16.26	31.27	38.97	22.45	52.43	60.50
Query2Doc	29.79	61.49	72.34	22.91	38.32	46.78	24.77	59.76	66.98
SeCom	37.02	64.04	77.45	16.11	28.54	34.74	23.08	53.49	61.17
A-Mem	35.40	63.60	73.60	21.25	37.81	46.32	31.15	62.65	71.49
MemoryOS	36.95	60.54	70.98	20.34	34.39	41.49	29.25	53.07	59.16
Ours	**50.21**	**77.66**	**84.68**	**28.30**	**45.67**	**53.47**	**49.01**	**71.35**	**79.63**
Base Retriever: QAMiniLM									
HyDE	31.91	61.06	73.19	17.27	30.56	37.61	30.62	65.68	74.17
Mill	35.32	63.19	74.04	9.42	18.98	26.13	36.15	67.27	74.77
Query2Doc	33.62	61.06	75.11	19.08	32.63	40.94	33.51	65.82	74.63
SeCom	41.06	71.06	81.28	15.56	28.80	36.60	23.82	55.29	63.57
A-Mem	32.40	61.00	71.60	20.49	37.51	45.17	30.83	62.68	70.75
MemoryOS	37.37	59.71	69.31	22.81	37.66	44.91	42.07	65.12	72.23
Ours	**49.79**	**76.17**	**84.47**	**27.14**	**44.86**	**52.87**	**50.39**	**72.83**	**80.23**

Table 8: The ablation study of the proposed classification strategy on four datasets. "w/ Classify" represents adopting the classification strategy to filter memory, "w/o Classify" denotes the opposite.

Method	Recall@1	NDCG@1	Recall@3	NDCG@3	Recall@5	NDCG@5	Recall@10	NDCG@10
LongMemEval-S								
w/o Classify	43.40	43.40	73.62	54.00	83.83	60.33	**94.26**	65.79
w/ Classify	**50.21**	**50.21**	**79.79**	**60.32**	**86.81**	**65.47**	92.34	**70.15**
LoCoMo								
w/o Classify	20.75	20.75	36.56	30.86	44.06	34.16	53.93	37.31
w/ Classify	**25.13**	**25.13**	**41.99**	**35.88**	**49.70**	**39.06**	**59.47**	**42.36**
LongMemEval-M								
w/o Classify	32.77	32.77	57.45	40.85	67.45	46.22	80.34	51.50
w/ Classify	**37.45**	**37.45**	**63.19**	**44.37**	**73.83**	**50.41**	**80.64**	**54.52**
PerLTQA								
w/o Classify	55.32	55.32	80.44	77.88	86.54	80.75	92.35	82.79
w/ Classify	**61.98**	**61.98**	**82.45**	**80.42**	**88.09**	**83.09**	**93.80**	**85.07**

D.2 Router Performance Ablation

A core design of MemoType is the memory type router that classifies both memories and queries into distinct types, enabling type-aware retrieval within a filtered subset rather than the full corpus. To verify whether this routing mechanism genuinely contributes to retrieval quality, we compare two settings: "w/ Classify", which filters the memory corpus based on predicted query types before retrieval, and "w/o Classify", which retrieves from the entire unfiltered corpus. Importantly, no retrieval enhancement strategies are applied in either setting, isolating the effect of classification filtering from other components.

As shown in Table 8, the classification strategy yields consistent improvements across all four datasets. On LongMemEval-S, filtering improves Recall@1 from 43.40 to 50.21 and NDCG@5 from 60.33 to 65.47, indicating that narrowing the search space helps the retriever focus on semantically relevant candidates. On LoCoMo, the gains are even more pronounced, with Recall@3 increasing from 36.56 to 41.99 and NDCG@10 rising by 5.05 points. On LongMemEval-M, the most challenging

Table 9: The ablation study of memory pruning strategy on three datasets. "w/ Prun" denotes adopting the proposed memory pruning strategy, and"w/o Prun" denotes the contrary.

Strategy	GPT4Judge	F1	BLEU	Rouge1	Rouge2	RougeL	RougeLsum	BERTScore
LongMemEval-S								
w/o Prun	55.60	19.55	4.48	21.32	10.17	19.54	19.70	85.03
w/ Prun	**56.60**	**20.31**	**4.91**	**22.18**	**10.70**	**20.28**	**20.32**	**85.22**
LoCoMo								
w/o Prun	48.39	20.30	4.11	21.36	10.63	19.93	19.91	85.55
w/ Prun	**49.19**	**22.23**	**4.64**	**23.28**	**11.83**	**21.83**	**21.86**	**85.80**
PerLTQA								
w/o Prun	62.58	45.69	19.27	49.61	31.84	43.23	43.24	91.10
w/ Prun	**62.83**	**47.26**	**20.65**	**51.20**	**33.02**	**44.78**	**44.79**	**91.61**
LongMemEval-M								
w/o Prun	49.60	17.78	4.05	19.25	8.61	17.41	17.50	84.62
w/ Prun	**50.20**	**18.77**	**4.24**	**20.36**	**9.19**	**18.56**	**18.64**	**85.02**

dataset with nearly 500 sessions per question, Recall@1 improves by 4.68 points, demonstrating that type-aware filtering becomes increasingly valuable as the corpus scale grows. On PerLTQA, Recall@1 increases from 55.32 to 61.98 with consistent gains across all metrics.

These results confirm that the router effectively reduces retrieval noise by eliminating irrelevant memory categories from the candidate pool. The improvement is achieved purely through corpus filtering without any query or memory augmentation, validating that accurate memory classification alone provides a strong foundation for downstream retrieval.

D.3 Memory Pruning Ablation

While the main paper presents the pruning ablation on F1 for two datasets, we provide comprehensive results across all four datasets and all generation metrics in this section. The motivation is to examine whether the query-conditioned pruning module consistently benefits generation quality, or whether its effect is limited to specific metrics or datasets.

As shown in Table 9, applying the pruning module yields consistent improvements across all datasets and nearly all metrics. On LoCoMo, the pruning strategy improves F1 from 20.30 to 22.23 and Rouge1 from 21.36 to 23.28, representing gains of approximately 1.93 and 1.92 points respectively. On PerLTQA, BLEU increases from 19.27 to 20.65 and RougeL improves from 43.23 to 44.78. On LongMemEval-M, pruning boosts F1 by nearly 1 point and RougeL by 1.15 points. Even on LongMemEval-S, where the baseline performance is already strong, pruning still provides noticeable gains across all metrics.

The consistent improvements across diverse metrics and datasets confirm that retrieved memory segments inevitably contain information irrelevant to the query, which introduces noise during response generation. The pruning module addresses this by adaptively retaining only query-relevant content before feeding memories into the LLM. Notably, the gains are more substantial on lexical overlap metrics such as BLEU and Rouge compared to BERTScore, suggesting that pruning primarily helps by removing distracting content that causes the model to generate off-topic tokens, while the overall semantic coherence is less affected.

D.4 Per-Type Retrieval Strategy Contributions

To investigate the independent contribution of each memory-type-specific retrieval strategy, we conduct fine-grained ablation experiments where only one strategy is activated at a time. Specifically, "Pure" denotes the baseline with classification filtering but no retrieval enhancement, "w/ GM" applies only the general semantic memory strategy, "w/ EM" applies only the episodic memory strategy, "w/ PM" applies only the personal semantic memory strategy, and "w/ All" activates all three strategies simultaneously.

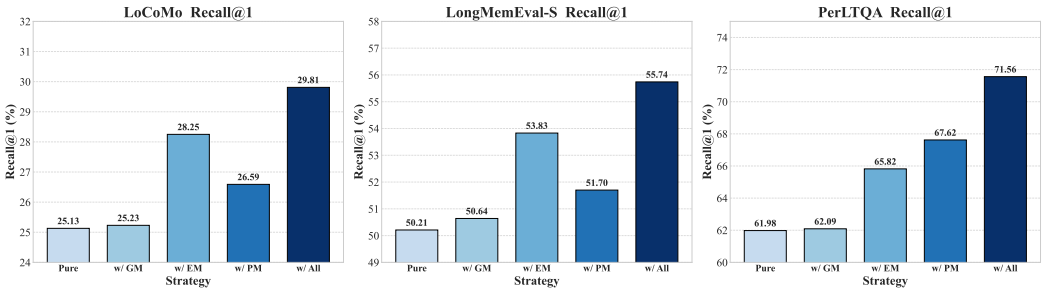


Figure 11: Strategy Ablation on Retrieval Recall@1 across Three Benchmarks.

As shown in Figure 11 and Figure 12, each individual strategy contributes positively to retrieval performance across all three datasets. Among the three strategies, the episodic memory strategy consistently yields the largest gains. On LoCoMo, w/ EM improves Recall@1 from 25.13 to 28.25, a gain of 3.12 points, while on LongMemEval-S, it achieves 53.83 compared to the Pure baseline of 50.21. This aligns with our expectation, as episodic memory involves temporal reasoning and structured event elements that benefit most from element-decomposed matching. The personal semantic memory strategy also provides substantial improvements, particularly on PerLTQA where Recall@1 increases from 61.98 to 67.62, reflecting the prevalence of user-specific queries in this dataset. The general semantic memory strategy yields relatively modest gains, which is expected since standard embedding similarity already handles factual knowledge reasonably well.

Crucially, combining all three strategies consistently outperforms any individual one, confirming that the strategies are complementary rather than redundant. The full combination achieves the highest scores across all datasets and metrics, validating the design principle of assigning tailored retrieval strategies to different memory types.

D.5 Router Architecture Ablation

Our default memory type router is built on Qwen3-1.7B fine-tuned with LoRA, which naturally raises the question of whether the strong routing performance stems from the model's inherent reasoning capacity or from the quality of the TriMEM training data. To disentangle these two factors, we replace Qwen3-1.7B with three lightweight encoder-only models, BERT, DeBERTa, and RoBERTa, all fine-tuned on the same TriMEM dataset, and evaluate the end-to-end retrieval performance.

As shown in Table 10, all three encoder-based routers achieve competitive results that are remarkably close to the Qwen3-1.7B router. On LongMemEval-S, BERT achieves a Recall@1 of 55.74, only 3.41 points below Qwen3-1.7B, while RoBERTa narrows this gap further to just 1.06 points with a Recall@1 of 58.09. On LoCoMo, BERT reaches 26.54 in Recall@1, and RoBERTa achieves 27.49, both delivering substantial improvements over the baselines reported in the main experiments. A similar pattern is observed on LongMemEval-M, where BERT attains 46.17 in Recall@1 and RoBERTa reaches 49.15, even slightly surpassing Qwen3-1.7B on this dataset.

These results lead to two important conclusions. First, the effectiveness of MemoType's routing mechanism is primarily driven by the quality and annotation precision of the TriMEM dataset rather than the scale or reasoning ability of the router model. Even BERT, with only 110M parameters and no generative capability, can serve as a competent router when trained on well-annotated data. Second, since encoder-only models are significantly faster at inference than generative LLMs, practitioners can opt for lightweight routers such as RoBERTa to achieve near-optimal performance with substantially lower latency, making MemoType more practical for deployment in real-world applications where efficiency is a critical concern.

D.6 Cost Analysis

We compare the LLM cost of memory construction on four benchmarks. A-Mem invokes the LLM twice per turn. HippoRAG2 invokes the LLM twice per chunk for named-entity and triple extraction; since each turn becomes a chunk, the cost grows linearly with the corpus. MemoType classifies every turn with a local Qwen3-1.7B router and only invokes the paid LLM on turns routed as

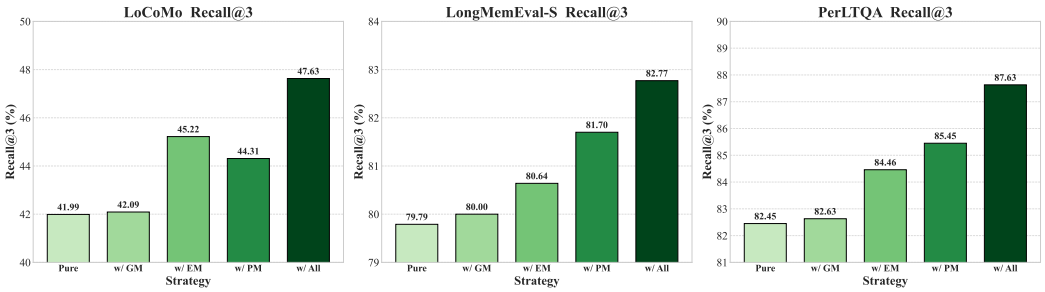


Figure 12: Strategy Ablation on Retrieval Recall@3 across Three Benchmarks.

Table 10: The Ablation Study of BERT Encoder.

Method	Recall@1	NDCG@1	Recall@3	NDCG@3	Recall@5	NDCG@5	Recall@10	NDCG@10
LongMemEval-S								
BERT	55.74	55.74	81.06	63.40	88.72	68.64	93.62	72.54
DeBERTa	55.32	55.32	81.91	63.19	89.36	68.90	95.74	72.62
RoBERTa	58.09	58.09	83.83	65.27	90.21	70.70	95.53	74.41
Qwen3-1.7B	59.15	59.15	83.19	66.74	90.21	71.60	92.77	74.85
LoCoMo								
BERT	26.54	26.54	43.15	36.77	50.60	39.79	61.48	43.56
DeBERTa	26.74	26.74	44.16	38.08	51.71	41.15	62.08	44.56
RoBERTa	27.49	27.49	43.96	37.46	51.31	40.52	62.13	44.27
Qwen3-1.7B	31.07	31.07	47.73	40.97	54.08	43.66	65.01	47.12
LongMemEval-M								
BERT	46.17	46.17	71.91	52.95	80.21	58.35	87.45	62.29
DeBERTa	46.38	46.38	68.94	49.73	77.66	55.55	86.81	59.91
RoBERTa	49.15	49.15	69.57	51.96	79.79	57.52	88.51	62.12
Qwen3-1.7B	48.09	48.09	70.43	52.20	78.09	57.09	84.89	61.27

Episodic Memory (EM) for event extraction, so its API cost scales with |EM| rather than with the full turn count. Table 11 reports the result. The fraction of EM turns is 29.9% on LoCoMo, 8.1% on LongMemEval-S, 2.2% on LongMemEval-M, and 80.9% on PerLTQA. MemoType is consistently the cheapest across all four datasets, and the gap widens as the corpus grows: on LongMemEval-M it reduces the API bill by $146\times$ over HippoRAG2 and $248\times$ over A-Mem. Even on PerLTQA, where most turns are EM and the gap is smallest, MemoType still halves the cost of HippoRAG2 and is more than $3\times$ cheaper than A-Mem. The local Qwen3-1.7B router is not billed but listed for transparency.

D.7 Empirical Evidence for Semantic Clustering

A central premise of Theorem 3.3 is the angular separation assumption (Assumption 3.1): different memory categories occupy distinct directions in the embedding space. To verify that this assumption is not merely a theoretical convenience but reflects a real property of dialogue memory corpora, we conduct a t-SNE visualization on two benchmarks. Specifically, we randomly sample 500 turns from each of the three memory categories (Episodic Memory, Personal Semantic Memory, and General Semantic Memory) on LongMemEval-S and PerLTQA, encode them with Contriever, and project the resulting embeddings into two dimensions via t-SNE [Wan et al., 2024b,a, Yu et al., 2023a,b].

As shown in Fig. 13, the three memory types form clearly identifiable clusters on both benchmarks, with each category concentrated in a distinct region of the embedding space. The separation is particularly pronounced between General Semantic Memory and the other two categories, while Episodic Memory and Personal Semantic Memory also exhibit visible boundaries despite their finer-grained semantic distinction. This empirical pattern directly corroborates Assumption 3.1: the centroids of different categories point in non-identical directions, with non-trivial angular separation $\delta > 0$.

Table 11: The cost analysis across four benchmarks. "Calls" is the number of paid LLM calls during memory ingestion. Due to the time limits, †HippoRAG2 on LongMemEval-M is linearly projected from $5/500$ samples.

Dataset	Method	Calls	Calls/sample	Prompt (M)	Compl. (M)	Total (M)	USD
LoCoMo	A-Mem	11,764	1,176.4	5.73	1.82	7.56	$1.95
	HippoRAG2	11,760	1,176.0	3.97	0.33	4.30	$0.79
	MemoType	1,756	175.6	0.29	0.08	0.37	**$0.09**
LongMemEval-S	A-Mem	493,574	987.1	326.67	76.50	403.18	$94.90
	HippoRAG2	492,146	984.3	254.34	29.21	283.56	$55.68
	MemoType	19,897	39.8	3.78	1.43	5.21	**$1.42**
LongMemEval-M	A-Mem	4,875,958	9,751.9	3,222.93	755.77	3,978.70	$936.90
	HippoRAG2†	≈4,909,400	9,818.8	2,523.91	283.99	2,807.90	$548.98
	MemoType	52,434	104.9	10.12	3.75	13.87	**$3.77**
PerLTQA	A-Mem	3,500	109.4	2.27	0.54	2.81	$0.67
	HippoRAG2	3,488	109.0	1.76	0.30	2.05	$0.44
	MemoType	1,416	44.2	0.49	0.19	0.68	**$0.19**

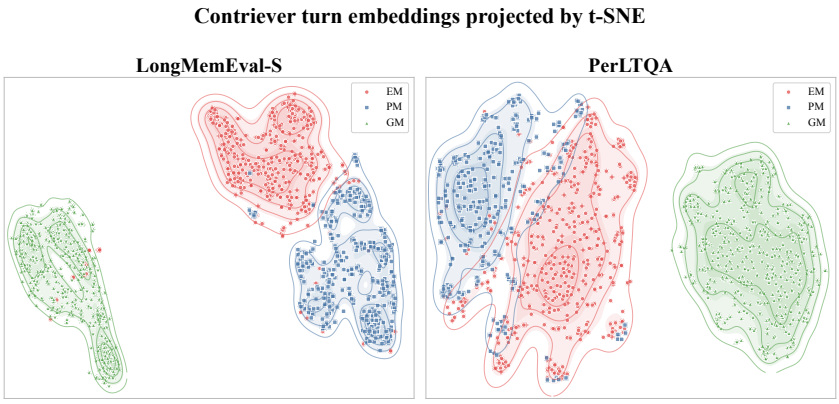


Figure 13: The Semantic Clustering Evidence with Contriever Embedding on two Benchmarks.

Consequently, the upper bound established in Theorem 3.3 is not vacuous on real corpora—a single shared encoder genuinely faces the geometric tension described in §3, motivating the type-aware retrieval design that processes each cluster through a tailored strategy.

E Extended Related Work

E.1 RAG-based Memory Methods

A growing body of work equips LLM agents [Xu et al., 2025a, Jiang et al., 2026, 2025c, Zhang et al., 2026a, 2025b, 2026b, Li et al., 2026] with an external memory store that is written, indexed, and retrieved at inference time, effectively casting long-term memory as a retrieval-augmented generation (RAG) problem. Early systems instantiate memory as a vector database of past turns or summaries and recall semantically relevant entries to condition generation, often with handcrafted forgetting or salience heuristics [Xu et al., 2023, Chen et al., 2025, Ong et al., 2025, Kim et al., 2024, Jiang et al., 2025a,b, Lu et al., 2023, Wang et al., 2023b, Zhang et al., 2026c, Xu et al., 2025b, Liu et al., 2026]. To better organize what is stored, knowledge-centric methods build structured notes or graphs that link concepts and reasoning traces across sessions, ranging from cached chains-of-thought [Liu et al., 2023] and agentic note networks [Xu et al., 2025c] to neurobiologically inspired hippocampal indexes [Gutiérrez et al., 2024]. A second line treats memory as a tiered architecture inspired by virtual memory or human cognition, paging information between fast working context and slower long-term stores under explicit read/write controls [Packer et al., 2023, Li et al., 2025, Kang et al., 2025, Fang et al., 2025a, Bae et al., 2022, Chhikara et al., 2025]. Reflection- and consolidation-based approaches instead periodically rewrite memory: generative agents distill experience into higher-

level reflections [Park et al., 2023], while subsequent work introduces prospective and retrospective consolidation, scalable extraction pipelines, or learned writing policies [Tan et al., 2025, Chhikara et al., 2025, Nan et al., 2025]. Across these families, the core RAG recipe remains the same—encode interaction history into a retrievable store and surface relevant items on demand—but they differ substantially in granularity (turn, session, fact, graph), update discipline (online, batched, sleep-time), and retrieval signal (embedding similarity, keywords, structural traversal).

E.2 RL-based Memory Methods

While most retrieval-augmented memory systems rely on hand-crafted write/update/retrieve heuristics, a recent line of work casts memory management itself as a sequential decision problem and trains the controller with reinforcement learning so that storage and recall are optimized for downstream task reward rather than surface-level similarity [Yan et al., 2025, Wang et al., 2025, Yu et al., 2025, Shi et al., 2025, Xu et al., 2026]. A first family learns explicit *memory-operation policies*: an LLM agent emits structured actions such as ADD, UPDATE, DELETE, and NOOP, and is fine-tuned with PPO or GRPO using outcome rewards from QA accuracy, often coupled with a separate RL-trained answering or retrieval agent [Yan et al., 2025, Yu et al., 2026]. A second family targets *end-to-end memory construction*: Mem-α trains a multi-component (core/episodic/semantic) memory with tool-style operations and generalizes from short training contexts to sequences an order of magnitude longer [Wang et al., 2025], while MemAgent extends RL to multi-conversation training with DAPO-style updates so the agent can operate on million-token histories [Yu et al., 2025]. A third strand focuses on *reward shaping and credit assignment*, since memory rewards are typically sparse and trajectory-level: ReMemR1 mixes final-answer rewards with dense step-level signals over revisitable memory [Shi et al., 2025]. These memory-centric RL methods are closely related to, but distinct from, RL-trained retrieval/search agents that learn when and how to query an external corpus [Jin et al., 2025, Song et al., 2025, Chen et al.], in that the policy must additionally decide *what to remember* and *when to forget*. Across the literature, the common message is that learned memory controllers consistently outperform fixed heuristics on long-horizon benchmarks such as LoCoMo, LongMemEval, and MSC.

F Prompts

Transfer Prompt for User-Assistant Conversation

System Role: You are an advanced text transformation AI. Your task is to process a given conversation between a user and an assistant.

Rules:

1. Replace specific nouns (e.g., occupations, professional fields, locations, dates, names, times, and objects) with alternative but relevant ones, ensuring the topic of the conversation remains unchanged.
2. Summarize both the user's and assistant's messages.
3. Limit user's responses to 40 words.
4. Limit assistant's responses to 80 words.
5. You can rearrange the order of the sentences to maintain better coherence between them.

Output Format:

```
[user]: [content]
[assistant]: [content]
```

Input: $\{text\}$

Figure 14: Transfer Prompt for User-Assistant Conversation

Transfer Prompt for User-User Conversation

System Role: You are an advanced text transformation AI. Your task is to process a given conversation between users.

Rules:

1. Replace specific nouns (e.g., names, occupations, professional fields, locations, dates, times, and objects) with alternative but relevant ones, ensuring the topic of the conversation remains unchanged.
2. You may try rearranging the order of the sentences to maintain better coherence between them.

Output Format:

```
[user name 1]: [content]
[user name 2]: [content]
```

Input: $\{text\}$

Figure 15: Transfer Prompt for User-User Conversation

Episodic Memory Classification Prompt

Goal: You are an advanced AI tasked with determining whether a specific piece of dialogue contains Episodic Memory. Use the following definition, guidelines, and examples to identify whether the provided dialogue qualifies as Episodic Memory.

Definition: Episodic Memory refers to specific episodes or events that occurred (are planned to occur, or the thoughts on future plans) at a particular place and time. For the dialogue to qualify as Episodic Memory, it often includes details:

When: A time, date, time frame, or vague contextual time expressions.
Where: A location or context.
Who: Participants.
What: Actions, or objects involved in the event.

Additional Guidelines:

- The time and location information does not always need to be precise. Vague or contextual expressions such as "when I was driving," "during lunch", or "while I was in school," can also qualify as time-related or location-related details if they provide sufficient information for the event.
- Questions that reveal or imply specific episodes or events information should also be classified as Episodic Memory.
- Missing one type of detail does not disqualify the dialogue from being Episodic Memory if the other details are sufficiently explicit to describe an instance.
- If the memory only describes general habits, vague knowledge, or patterns, it does not qualify as Episodic Memory.
- If the dialogue contains at least one instance of Episodic Memory, the output must be "Episodic Memory", regardless of whether other parts of the dialogue do not contain Episodic Memory.

Output Requirements:

`If contains Episodic Memory` $\rightarrow$ `Episodic Memory`
`If does not contain Episodic Memory` $\rightarrow$ `None`
Do not include additional context, comments, symbols, or explanations.

Input: $\{text_to_be_classified\}$

Output: Now, based on the above instructions, determine whether the input data contains Episodic Memory.

Figure 16: Episodic Memory Classification Prompt

Personal Semantic Memory Classification Prompt

Goal: You are an advanced AI tasked with determining whether a segment of dialogue contains Personal Semantic Memory. Use the following definition, guidelines, and examples to identify whether the provided dialogue qualifies as Personal Semantic Memory.

Definition: Personal Semantic Memory is a form of long-term memory consisting of knowledge specifically related to the person (user or another individual mentioned in the dialogue), such as descriptions of their information, preferences, habits, thoughts, background, emotions, education or work career-related information, and any other knowledge that reflects the individual's identity.

Additional Guidelines:

- Content that reveals personal information, preferences, habits, plans, background information, emotions, personal history, education or work career-related information, or any other knowledge that related to the individual's identity must be classified as Personal Semantic Memory.
- Statements that express the individual's intentions, work arrangements, or preliminary plans (not include details), even if they are inferred, belong to the category of Personal Semantic Memory.
- General knowledge unrelated to personal information, preferences, or emotions do not qualify as Personal Semantic Memory.
- Questions that reveal or imply personal details, preferences, or habits (user or another individual), should also be classified as Personal Semantic Memory.
- If the dialogue contains at least one instance of Personal Semantic Memory, the output must be "Personal Semantic Memory", regardless of whether other parts of the dialogue do not contain Personal Semantic Memory. The unanswered questions or requests in the conversation should not affect the final classification result.

Output Requirements:

`If contains Personal Semantic Memory` $\rightarrow$ `Personal Semantic Memory`
`If does not contain Personal Semantic Memory` $\rightarrow$ `None`
Do not include additional context, comments, symbols, or explanations.

Input: $\{text_to_be_classified\}$

Output: Now, based on the above instructions, determine whether the input data contains Personal Semantic Memory.

Figure 17: Personal Semantic Memory Classification Prompt

General Semantic Memory Classification Prompt

Goal: You are an advanced AI tasked with determining whether a given text contains General Semantic Memory based on the following definition, criteria, and examples.

Definition: General Semantic Memory is a form of long-term memory consisting of non-personalized knowledge about the world, including facts, concepts, language, universally applicable methods, and objective descriptions of locations or objects. This includes information that is not tied to any specific individual or event. For information to qualify as General Semantic Memory, it must meet at least one of the following criteria:

- Non-Personalized Knowledge or Broadly Applicable Suggestions: It could include general knowledge, factual information, or widely applicable suggestions and methods. Examples: Scientific facts, historical events, cultural concepts, general knowledges, or general advice on common activities.
- Objective Descriptions of Locations, Objects, or Concepts: Information that provides objective, factual details about locations, objects, or entities. Examples: Descriptions of landmarks, places, or any items.

Additional Guidelines:

- Focus on analyzing sentences that provide valid information. Dialogue that consists solely of questions without answers should not be used as a basis for General Semantic Memory classification.
- The information about a specific event or personalized to the user (e.g., user preferences, personal habits, or individual-specific details) does not qualify as General Semantic Memory.
- If the dialogue contains at least one instance of General Semantic Memory, the output must be "General Semantic Memory", regardless of whether other parts of the dialogue do not contain General Semantic Memory.

Output Requirements:

`If contains General Semantic Memory` $\rightarrow$ `General Semantic Memory`
`If does not contain General Semantic Memory` $\rightarrow$ `None`
Do not include additional context, comments, symbols, or explanations.

Input: $\{text_to_be_classified\}$

Output: Now, based on the above instructions, determine whether the input data contains General Semantic Memory.

Figure 18: General Semantic Memory Classification Prompt

Memory Multi-Classification Prompt

Goal: You are an advanced AI tasked with classifying a dialogue between a user and a chatbot into one or more of the following memory types: "Episodic Memory," "Personal Semantic Memory," or "General Semantic Memory." Use the following definitions to guide your classification:

- Episodic Memory: A form of long-term memory consisting of non-personalized knowledge about the world, including facts, concepts, language, universally applicable methods, and objective descriptions of locations or objects. This includes information that is not tied to any specific individual or event.
- Personal Semantic Memory: A form of long-term memory consisting of knowledge specifically related to the person (user or another individual mentioned in the dialogue), such as descriptions of their information, preferences, habits, thoughts, background, emotions, education or work career-related information, and any other knowledge that reflects the individual's identity.
- General Semantic Memory: A form of long-term memory consisting of general knowledge about the world, including facts, concepts, language, universally applicable methods, and objective description. This includes information that is broadly relevant, commonly recognized, or widely recommended.

Output Requirements: Your output must strictly adhere to one or more of the following memory types, separated by commas if multiple apply:

`Episodic Memory`, `Personal Semantic Memory`, `General Semantic Memory`
`If none of the above types apply` $\rightarrow$ `None`
Do not include any additional words or context beyond the specified output format.

Carefully analyze the dialogue and determine the appropriate memory type(s) based on the content and context provided.

Input: $\{text_to_be_classified\}$

Output: Now, classify the memory type(s) of the input data based on the instructions above.

Figure 19: Memory Multi-Classification Prompt

HyDE Prompt

Please write a paragraph that answers the question.

Question: $\{query\}$

Output:

Figure 20: HyDE Prompt

MILL Prompt

What sub-queries should be searched to answer the following query? Please generate 5 sub-queries with their related passages.

Question: $\{query\}$

Only present the subquestion, without any other words and explanation.

Figure 21: MILL Prompt

Query2Doc Prompt

Write a passage that answers the given query:

Question: $\{query\}$

Output:

Figure 22: Query2Doc Prompt

Answer Generation Prompt for User-Assistant Conversation

Goal: Suppose a user has recent conversation records with assistant. Use your imagination to generate conversation records. The generated conversation records must contain information related to the given query. The conversation must be logically clear and structurally reasonable, representing a discussion on a specific topic rather than a direct recollection of the query. Conversation record should be generated from one of the following perspectives:

- When the query is not an advice-seeking type of question, a sentence can be output that directly provides an answer matching the query, avoiding expressions other than stating the answer.
- Instead of providing a direct answer, it can describe the user's preferences, habits, events, or background related to the topic of the query. This is not a direct answer to the query, but should be an additional statement from the user, especially for advice-seeking type queries.

Input Data: The input is a natural language query posed by the user, typically related to the previous conversations information.

Output Requirements:

- The output must not exceed one sentence. You should determine whether the query-related information is mentioned by the user or the assistant.
- Ensure that the response contains only the dialogue content of one speaker.

Input: $\{query_to_be_answered\}$

Output: Generate a hypothetical dialogue record.

Figure 23: Answer Generation Prompt for User-Assistant Conversation

Answer Generation Prompt for User-User Conversation

Goal: Suppose there is a conversation between users. Imagine you are the user mentioned in the query and generate a dialogue in the first person. The generated dialogue record must contain relevant information that answers the given query. The memory must be logical, well-structured, and tailored to the type of question provided.

Input Data: The input is a query in natural language that asks a specific question or seeks information.

Output Requirements:

- Completeness: Ensure that each dialogue memory provides sufficient information to address the query comprehensively.
- Length: The output must not exceed one to two sentences. Ensure brevity while maintaining clarity and relevance.
- Perspective: The response should be written from the perspective of the user mentioned in the query, as a single speaker.

Output Format:

```
[user name in query]: [response content]
```

Input: $\{query_to_be_answered\}$

Output: Generate a hypothetical dialogue record.

Figure 24: Answer Generation Prompt for User-User Conversation

Hypothetical Query Generation Prompt

Instruction: You are reading one short turn from a dialogue, either between two people or between a user and an AI assistant. The turn typically conveys a personal fact about a participant (e.g., identity, family, work, places, hobbies, preferences, plans, or experiences). Imagine that, later on, someone wants to recall this exact turn by issuing a search query to their memory. Write $\{K\}$ different short questions whose best answer would be exactly this turn.

Guidelines:

- Cover different aspects of the turn (who, what, where, when, why, how), such as the person, the topic, what was stated or done, and the time or place if mentioned.
- Each question must stand on its own and must not assume the reader has already seen the turn.
- Use the participant's name if it appears in the turn. If the turn is the user speaking to an assistant, phrase the question in the first person ("I", "my", "me"), as the user asking about themselves rather than asking for general knowledge.
- Keep each question short and concrete. Do not invent facts that are not in the turn, and avoid generic filler such as "What did the speaker say next?".

Output Format: Output exactly $\{K\}$ lines, one question per line. No numbering, no quotation marks, and no extra commentary.

Input: Turn: $\{text\}$

Figure 25: Hypothetical Query Generation Prompt

Conversation Event Extract Prompt

Instruction: Identify and extract the elements of events mentioned in the text below. The text was recorded on **{session_date}**. If the dataset does not record a session date this field is filled with the literal string "unknown"; in that case skip any relative-time resolution.
The text may be either (a) a turn from a chat between a user and an AI assistant, or (b) a piece of structured personal-knowledge content (profile, social relationships, education, profession, . . .). Many inputs do **not** describe a real event at all (chit-chat, opinions, task requests, stable attributes); for those cases follow the strict N/A rules below.

Output Format: Use exactly the following structured format. Repeat the block once per distinct event, separated by a blank line. If nothing event-like is present, emit one block with all three fields = 'N/A'.

```
[Event n]
Time:  [Absolute date in YYYY-MM-DD, YYYY-MM, YYYY, or 'N/A']
Person(s):  [Person(s) involved, or 'N/A']
Location:  [Location, or 'N/A']
```

When to Output Everything 'N/A': Emit a single all-N/A block when the input is:

- A question to the assistant ("How do I . . . ?", "What is . . . ?")
- A request to the assistant ("Write me a poem about . . . ")
- A generic opinion or feeling not tied to an event ("I think Python is great")
- A stable preference or habit ("I usually drink coffee in the morning")
- A hypothetical / future plan with no anchored time

Time — Strict Rules:

- Resolve every relative time expression ("yesterday", "last week", "two days ago", "this morning", "next month", . . .) against {session_date} and output the absolute date. Never keep the relative phrase in Time.
- If only year + month is clear, output `YYYY-MM`; year only, output `YYYY`; otherwise 'N/A'.

Person(s) — Strict Rules:

- Replace every literal "User"/"user" with "I".
- Never include "Assistant", "AI", "Bot", "ChatGPT", "the model", or any other non-human dialog role.
- Copy named third persons verbatim. Never invent names.

Location — Strict Rules: Fill only when an explicit place is mentioned in the text.

Conversation Date: {$session_date$}

Input Data: {$text_to_be_processed$}

Figure 26: Conversation Event Extract Prompt

Query Event Extract Prompt

Instruction: Identify and extract the event- or attribute-related elements mentioned in a query. The query is asked on **{question_date}**. If the dataset does not record a query date this field is filled with the literal string "unknown"; in that case skip any relative-time resolution and treat Time accordingly.
A query may target either (a) a concrete event referenced in the dialog (a trip, a graduation, an action the user took, a plan the user has), or (b) a stable personal attribute about the speaker or a third person named in the query (occupation, hobby, residence, relationship, etc.). Both cases are handled with the same four output fields below.

Output Format (exactly four lines, in this order):

```
Time: [Absolute date in YYYY-MM-DD, YYYY-MM, YYYY, or 'N/A']
Person(s): [Person(s) involved, or 'N/A']
Location: [Location, or 'N/A']
Event: [Concise event or attribute description]
```

Time — Strict Rules:

- Resolve every relative time expression ("yesterday", "last week", "two days ago", "this morning", ...) against {question_date} and output the absolute date. Never keep a relative phrase in Time.
- Output 'N/A' if the query, combined with the question date, does not pin down a specific date / month / year.
- Never invent a date that is not stated or implied by the query.
- For pure attribute questions (e.g., "What is X's occupation?"), Time is almost always 'N/A'.

Query Date: $\{question_date\}$

Input Data: $\{text_to_be_processed\}$

Figure 27: Query Event Extract Prompt

Memory Pruning Prompt

Role: You are a dialog assistant given a numbered, dated list of memories and a user query. You will (a) answer the query using only information that can be supported by the memories, and (b) explicitly cite the memory indices that you actually used.

Inputs:

- Query date: The date on which the user asks the question. May be the literal string "unknown" if the dataset does not record a query date; in that case skip any temporal reasoning.
- Question: The user query.
- Memory: A numbered list. Each item starts with the header `[Memory <i>] (session_date=<date>)` followed by the original dialog text. `session_date` may also be "unknown".

Procedure:

1. Read every memory carefully. When the question is time-sensitive (e.g., "last week", "yesterday", "since May"), use the per-memory `session_date` together with the Query date to ground relative temporal expressions before deciding which memories support the answer.
2. Produce a single concise answer using ONLY information that is supported by the memories. If no memory supports the question, answer "I don't know" rather than fabricating.
3. List the 1-based memory indices that you actually relied on, taken verbatim from the `[Memory <i>]` headers.

Output Format (exactly two lines, nothing else):

```
Answer: <your concise answer>
References: [i, j, k]
```

Rules for References:

- Use a JSON-style list of integers, e.g., `References: [1, 4]` or `References: []` when no memory supports the answer.
- Never invent an index that is not present in the Memory list.
- Do not include duplicate indices and do not wrap them in quotes.

Query date: $\{query_date\}$
Question: $\{question\}$
Memory: $\{context\}$

Figure 28: Memory Pruning Prompt

Response Prompt

Role: You are an intelligent dialog bot. You are shown a small set of Related Evidences (each annotated with a `session_date`) and the user's query (annotated with a Query date). The evidences have already been filtered down by an earlier reference-selection step, so every item below is expected to be relevant.

Read, understand, and synthesize the evidences, then output ONE concise, coherent and helpful answer. Provide the answer text directly, without prefixes, prologues, or any extra commentary.

Query date: $\{query_date\}$
Context: $\{context\}$
Question: $\{question\}$

Figure 29: Response Prompt, which follows [Lu et al., 2023, Pan et al., 2025]

GPT Judge Prompt

I will give you a question, a reference answer, and a response from a model. Please answer [[yes]] if the response contains the reference answer. Otherwise, answer [[no]].
If the response is equivalent to the correct answer or contains all the intermediate steps to get the reference answer, you should also answer [[yes]]. If the response only contains a subset of the information required by the answer, answer [[no]].

[User Question]
$\{question\}$
[The Start of Reference Answer]
$\{answer\}$
[The End of Reference Answer]
[The Start of Model's Response]
$\{response\}$
[The End of Model's Response]

Is the model response correct? Answer `[[yes]]` or `[[no]]` only.

Figure 30: GPT Judge Prompt, which follows [Wu et al., 2025, Zheng et al., 2023]

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims are stated in the abstract and introduction

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We have given the full assumptions and complete proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have included all the information needed to reproduce the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have released the code. In addition, the datasets used in our work are all public datasets.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to the Implementation Details Section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We conduct the two-sided t-test on our results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to Implementation Details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We conform to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the impacts in the Introduction section.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: All datasets used in our work are public datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite the dataset and models used in our work.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have given the descriptions in our paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing experiments are conducted in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No potential risk is found in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLM usage does not impact the core methodology, scientific rigorousness, or originality of the research.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.