Flexibly Utilize Memory for Long-Term Conversation via a Fragment-then-Compose Framework

Cai Ke[1,2], Yiming Du[3,4], Bin Liang[3,4]*, Yifan Xiang[3], Lin Gui[5], Zhongyang Li[6], Baojun Wang[6], Yue Yu[2], Hui Wang[2]*, Kam-Fai Wong[3,4], and Ruifeng Xu[1,2]*

[1]Harbin Institute of Technology, Shenzhen, China [2]Peng Cheng Laboratory, China
[3]The Chinese University of Hong Kong, Hong Kong, China
[4]MoE Key Laboratory of High Confidence Software Technologies, CUHK, China
[5]King's College London, UK [6]Huawei Noah's Ark Lab, China
kecai@stu.hit.edu.cn, xuruifeng@hit.edu.cn

Abstract

Large language models (LLMs) have made significant breakthroughs in extracting useful information from conversation history to enhance the response in long-term conversations. Summarizing useful information from historical conversations has achieved remarkable performance, which, however, may introduce irrelevant or redundant information, making it difficult to flexibly choose and integrate key information from different sessions during memory retrieval. To address this issue, we propose a Fragment-then-Compose framework, a novel memory utilization approach for long-term open-domain conversation, called FraCom. To be specific, inspired by the concept of proposition representation from Cognitive Psychology, we first represent the conversation history as a series of predicates plus arguments for propositional representation to preserve key information useful for memory ("**Fragment**"). Then, we compose propositional graphs for the conversation history based on the connection between shared arguments ("**Compose**"). During retrieval, we retrieve relevant propositions from the graph based on arguments from the current query. This essentially allows for flexible and effective utilization of related information in long-term memory for better response generation towards a query. Experimental results on four long-term open-domain conversation datasets demonstrate the effectiveness of our FraCom in memory utilization and its ability to enhance response generation for LLMs.

1 Introduction

The remarkable advances in Large Language Models (LLMs) have led to the rapid development of open-domain conversations (Li et al., 2017; Zhang et al., 2018; Dinan et al., 2018; Rashkin et al., 2019; Baumgartner et al., 2020). By modeling and understanding historical dialogue information, LLMs have demonstrated strong response generation capabilities in open-domain conversations. Despite the remarkable progress made in open-domain conversation, when dealing with long-term conversations, LLMs still struggle to achieve satisfactory human-like interactions due to their lack of long-term memory (Xu et al., 2022a; Shi et al., 2023; Du et al., 2024; Zhang et al., 2024; Li et al., 2024; Levy et al., 2024; Liu et al., 2024).

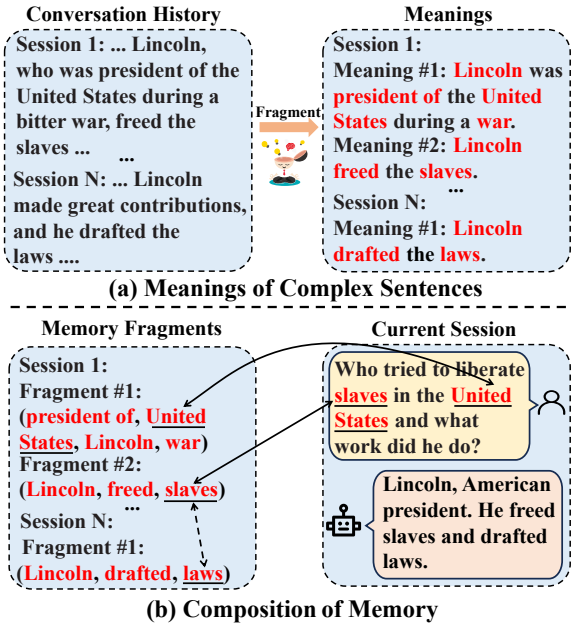


Figure 1: Examples of humans storing and utilizing memories from historical conversations based on Cognitive Psychology.

Memory is an essential aspect of human-like communication, which plays a pivotal role in sustaining long-term, high-quality interactions during conversations. By fusing memory, LLMs can generate more coherent, natural, and contextually relevant responses by effectively storing and recalling previous conversational data. This enhancement significantly improves the engagement, human likeness, and memorability of interactions. Recent

* Corresponding authors.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 21131–21148
November 4-9, 2025

studies focus on compressing historical information into summaries as memories with remarkable results (Xu et al., 2022a; Bae et al., 2022; Jang et al., 2023; Zhang et al., 2023; Lu et al., 2023; Zhong et al., 2024; Li et al., 2025; Ong et al., 2025; Chen et al., 2025; Wang et al., 2025). However, they are prone to trigger the influx of redundant or irrelevant information when producing memory from the historical conversation; not to mention that summarizing different sessions of historical conversation separately increases the difficulty of leveraging the relationship between core information of different sessions when retrieving memory for the query.

Based on Cognitive Psychology, humans break down a complex sentence into a series of simple sentences for memorization, which contain basic meaning about historical content (Schank, 1980; Tulving, 1983, 2002; Anderson, 2005; Yadav et al., 2022). As shown in Figure 1 (a), people tend to remember simple sentences "*Lincoln was president of the United States during a war.*" and "*Lincoln freed the slaves.*" for the complex conversation content. In addition, according to propositional analyses, people remember a complex sentence as a set of abstract meaning units that represent the simple assertions in the sentence (Kintsch, 1974, 2014). As shown in Figure 1 (b), words or phrases "*president-of*", "*United States*", "*Lincoln*", and "*war*" can preserve the meaning of the sentence. Especially in long-term memory, these words/phrases can be better remembered and used to restore memory. Meanwhile, these key words/phrases can also be flexibly combined and used during memory replay according to actual needs. Therefore, we argue that fragmenting the conversation history based on key information and flexibly composing memory regarding the query can lead to improved memory utilization in long-term conversations.

To reach this goal, we propose a Fragment-then-Compose framework (`FraCom`) to flexibly utilize memory, aiming at producing better responses for long-term conversation. Specifically, for the **Fragment** step, we get inspiration from proposition representations that "*a proposition is the smallest unit of knowledge that can stand as a separate assertion*" (Anderson and Bower, 1974), and prompt LLMs to obtain predicates and arguments as key words/phrases for each utterance in historical sessions, called *Memory Fragments*. This provides the basic material for flexibly composing memory in memory utilization according to the demand. Further, inspired by the representation of meaning in memory (Kintsch, 1974, 2014), for the **Compose** step, we leverage predicates and arguments as nodes to construct a propositional graph, where propositions with the same argument in different utterances and different sessions can be connected to obtain long-term memory for conversation history. Based on this, we get inspiration from Plausible Retrieval (Reder, 1982; Reder and Ross, 1983) and match the arguments of the current query/utterance from the propositional graph to capture the exact and plausible propositions, which constitute the retrieved memory information. This essentially allows the model to flexibly use memory based on key information of the current utterance, rather than using all memory for response. Experimental results on four long-term conversation datasets show that our `FraCom` can enhance the ability of memory utilization for LLMs in long-term conversations, thereby leading to improved response generation. Furthermore, we propose new memory metrics to evaluate memory usage and capacity. The results show that our method can achieve better and more efficient memory utilization compared to baselines.

The contribution of this work can be summarized as follows:

1) We explore a new paradigm to flexibly use memory for long-term conversations, which can save storage space while utilizing memory more effectively.

2) We are the first to fragment historical information and then compose a memory graph, which endows the model with the ability to retrieve long-term memory based on the key information in the current utterance.

3) Experimental results show that our method outperforms strong baselines in both response generation and memory utilization.

2 Related Work

Long-Term Conversations. Long-term conversation reflects real-world conversational scenarios and can achieve long-term interaction. Previous works focus on selecting valuable information on conversations to train models for generation (Bae et al., 2022; Xu et al., 2022a,b; Jang et al., 2023). A current trend is to build memory banks (Lu et al., 2023; Zhang et al., 2023; Zhong et al., 2024; Chen et al., 2025; Li et al., 2025; Ong et al., 2025; Wang et al., 2025) as a plug-and-play module for LLMs.

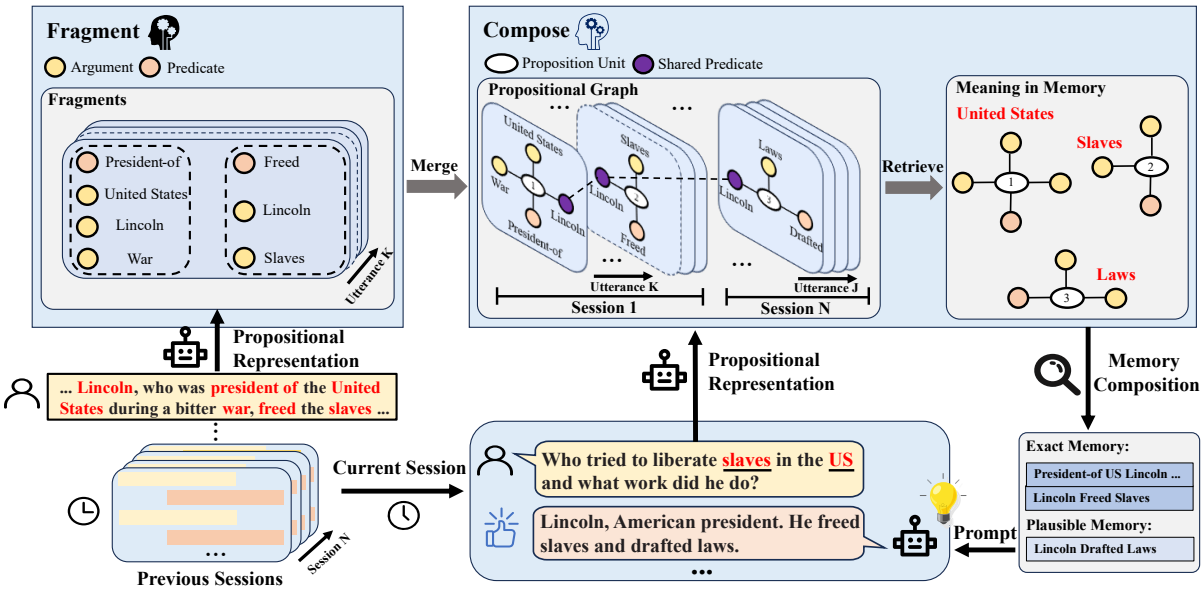


Figure 2: Illustration of our Fragment-then-Compose memory utilization framework.

Zhong et al. (2024), Chen et al. (2025) and Li et al. (2025) compress sessions into summaries and user-specific facts. Wang et al. (2025) summarize the conversation of each session and iteratively update the old memory. Moreover, Zhang et al. (2023) and Ong et al. (2025) pay more attention to the impact of time interval on generation. **Different from these methods using summaries for memory, we propose a new paradigm to flexibly use memory by fragmenting and composing key information and thus improve the performance of response generation and memory utilization**.

Cognitive Memory Modeling. Human memorization can be described as a fragmentation paradigm rather than a compressed summary (Schank, 1980; Tulving, 1983, 2002; Yadav et al., 2022). This means that humans only remember tiny fragments of what they experienced. In particular, these fragments tend to store the most meaningful information while often ignoring details that are considered less important (Anderson, 2005). The propositions (Weisberg, 1969; Anderson and Bower, 1974; Ratcliff and McKoon, 1978; Kintsch, 2014) in Linguistics are regarded as the smallest units of meaning and are the basic components of memory. Therefore, each memory fragment can be connected to a proposition unit. The recall process involves the memory retention (Nelson, 1971; Wickelgren, 1975; Nelson, 1978) and plausible retrieval (Reder, 1982; Reder and Ross, 1983) of these memory fragments, forming a coherent memory by composing related propositions, providing a foundation for flexible use of memory. **Different from knowledge graph-type work (Edge et al., 2024), our `FraCom` can preserve the logical structure of natural language to obtain language-driven propositional graphs through propositional representation**.

3 Methodology

In this section, we describe our proposed Fragment-then-Compose (`FraCom`) framework for long-term conversation in detail. Given a historical conversation $S = \{s_1, s_2, ..., s_N\}$, the long-term conversation aims to generate an accurate response u^* to a current utterance $U = \{u_1, u_2, ..., u_n\}$ by effectively leveraging memory derived from S. As shown in Figure 2, our `FraCom` mainly consists of three modules: 1) **Fragment module**, which extracts the key words or phrases in each utterance to obtain the memory fragments; 2) **Compose module**, which composes a propositional graph for the conversation history by linking the shared memory fragments; 3) **Memory Retrieval and Response Generation**, which retrieves and utilizes relevant memory meanings from the propositional graph based on the key information for response generation.

3.1 Fragment Module

For the **Fragment** step, we perform propositional representation (Anderson and Bower, 1974) to fragment each utterance in previous sessions to ob-

tain the memory fragments that retain the gist of memory. Specifically, we ask the LLMs to deduce with the prompt: "*The following is the conversation content: [Conversation]. Please extract the basic proposition (predicate, argument) from each sentence. According to the theory of Propositional Representation, 'predicate' usually corresponds to verbs, adjectives, and other predicates, while 'argument' usually corresponds to nouns. 'predicate' refers to the connection between the 'argument' referred to by these nouns*", where *[Conversation]* denotes historical sessions. As shown in Figure 2, given utterance u "*Lincoln, who was president of the United States during a bitter war, freed the slaves...*" will be fragmented into arguments (*United States, Lincoln, War*, etc.) and predicates (*President-of, Freed*, etc.). These arguments $A = \{a_1, a_2, ..., a_m\}$ and predicates $R = \{r_1, r_2, ..., r_n\}$ can form different propositions $P = \{p_1, p_2, ..., p_n\}$ in each session. Each p_i contains a proposition unit c_i, a predicate r_i, and the related arguments a_i, which can be composed into a propositional graph. Prompting details are depicted in Appendix K.

3.2 Compose Module

For the **Compose** step, we compose the above fragments as a dynamic propositional graph $G = (V, E)$ that incrementally incorporates propositions from each session by connecting the same argument between the original propositional graph and the new session. For each proposition p_i, we construct a propositional subgraph $G_i = (V_i, E_i)$:

$$V_i = \{c_i, r_i, a_{i1}, ..., a_{im}\}, \quad (1)$$

where a_{ij} represents the j-th argument in the i-th subgraph G_i and m means the number of arguments in p_i. The edge E_i connects c_i to r_i and a_{ij}, which is defined as follows:

$$E_i = \{(c_i, r_i)\} \cup \{(c_i, a_{ij}) \mid j = 1, ..., m\}. \quad (2)$$

A predicate node r_i and argument nodes a_{ij} connected to c_i can express a proposition p_i. For each session s, we connect the shared argument nodes of each proposition p_i to merge them into a session propositional graph G_s. When processing a new session t, we merge its propositional subgraphs G_t into G_s, which is defined as follows:

$$V \leftarrow V_i^t \cup V_i^s, \quad (3)$$

$$E \leftarrow E_i^t \cup E_i^s, \quad (4)$$

where shared arguments across propositions are connected through identity edges. The final propositional graph G preserves all historical propositions while maintaining long-term memory through shared argument nodes. The procedure for updating the graph with new propositions is detailed in Algorithm 1.

Algorithm 1: Propositional Graph Update

Input: G_{hist}: The existing graph
P_{new}: List of new propositions
Output: Updated graph G_{hist}

foreach *proposition* p_i *in* P_{new} **do**
  // p_i has predicate r_i and arguments A_i
  Create new proposition node c_i;
  Add c_i to G_{hist} as `proposition`;
  if *predicate* r_i *not in* G_{hist} **then**
    Add r_i to G_{hist} as `predicate`;
  Add edge (c_i, r_i);
  foreach *argument* a_{ij} *in* A_i **do**
    if *argument* a_{ij} *not in* G_{hist} **then**
      Add a_{ij} to G_{hist} as `argument`;
    Add edge (c_i, a_{ij});
return G_{hist}

3.3 Memory Retrieval and Response Generation

Inspired by Reder (1982); Reder and Ross (1983), we employ plausible retrieval to flexibly recall meaning in memories like humans. For the current session (starting from the second session), we perform propositional representation on current utterances to obtain arguments in propositions. For plausible retrieval, we use cosine to measure the similarity between the stored propositional graph G and each argument a_i in the utterance:

$$BertSim(a_i, G) = cos(E(a_i), E(G)), \quad (5)$$

where $E(\cdot)$ refers to Sentence-BERT encoder (Reimers, 2019). In fact, we only retrieve the argument nodes in G for similarity to determine whether the proposition is matched. Moreover, we set the similarity threshold θ to perform plausible retrieval. When $BertSim(\cdot) \geq \theta$, we regard the proposition p_i as a retrieved memory. Each proposition represents a sentence of meaning in memory. Finally, we can obtain proposition $\hat{P} = \{\hat{p_1}, \hat{p_2}, ..., \hat{p_n}\}$ for response generation.

Backbone	Methods	CC			MSC			GC			LME		
		B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert
Qwen2.5	Context	0.40	13.87	30.01	0.10	13.27	38.26	0.76	13.23	28.37	3.21	18.06	64.73
	Rsum	0.41	14.17	30.75	0.12	13.19	39.16	0.69	12.27	31.12	2.94	18.11	63.35
	MemoChat	0.19	11.22	27.20	0.08	11.09	35.61	0.53	10.58	29.41	**4.92**	**20.16**	64.78
	MemoryBank	0.35	14.19	30.33	0.13	13.30	**39.71**	0.74	12.67	31.20	2.87	18.29	64.35
	COMEDY	0.17	11.35	29.04	0.10	11.20	36.75	0.45	10.18	29.53	3.86	18.34	59.01
	Ours	**0.44**	**15.91**	**37.03**	**0.19**	**13.66**	38.76	**0.80**	**13.70**	**35.01**	3.98	19.83	**69.51**
Llama3	Context	0.29	11.43	28.46	0.10	11.52	32.91	0.35	11.21	29.16	4.51	15.37	47.64
	Rsum	0.36	12.98	29.56	0.11	12.37	36.27	0.48	11.32	**30.91**	3.42	17.74	59.67
	MemoChat	0.26	11.62	27.01	0.08	11.78	35.22	0.37	10.65	29.87	4.93	19.75	65.42
	MemoryBank	0.31	12.35	29.13	0.09	12.15	35.97	0.34	10.66	28.25	4.50	19.72	68.39
	COMEDY	0.14	10.48	27.33	0.09	10.33	34.59	0.30	8.99	28.59	3.79	17.27	57.34
	Ours	**0.51**	**14.91**	**30.13**	**0.12**	**12.90**	**36.92**	**0.56**	**11.51**	30.45	**5.51**	**20.65**	**69.59**
ChatGPT	Context	0.62	15.44	34.75	0.16	14.50	39.32	0.61	11.39	31.86	1.42	16.51	68.03
	Rsum	0.41	13.66	31.00	0.12	11.83	36.69	0.70	12.16	31.50	1.73	16.51	68.87
	MemoChat	0.70	15.71	30.25	0.14	13.44	36.02	0.80	13.04	31.02	1.85	16.48	64.74
	MemoryBank	0.23	10.60	22.67	0.07	10.09	31.04	0.50	10.13	25.38	1.72	16.70	68.35
	COMEDY	0.29	12.64	31.58	0.11	12.25	38.00	0.68	11.81	32.18	**2.56**	17.74	64.41
	Ours	**0.72**	**17.31**	**39.93**	**0.22**	**14.83**	**41.45**	**1.30**	**15.31**	**37.52**	2.30	**18.04**	**70.07**

Table 1: Automatic evaluation (%) of generation performance per episode. "**Bold Font**" means the highest results, while "Underlined Font" means second-highest results. "Context" denotes feeding history information directly into the long context of LLMs. *B-4 = BLEU-4, R-L = ROUGE-L, and Bert = BertScore. Appendix E for more results.

After memory retrieval, the response u^* is generated by LLMs that integrates each current utterance u_i and retrieved proposition $\hat{P}$ in current session. The optimal response is obtained by maximizing the conditional probability distribution:

$$u^* = \underset{u \in \mathcal{R}}{argmax}\ P_{LLM}(u|u_i, \hat{P}), \tag{6}$$

where $\mathcal{R}$ represents a set of all possible responses.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate our method on four long-term multi-session conversation datasets: **Conversation Chronicles** (CC) (Jang et al., 2023), **Multi-Session Chat** (MSC) (Xu et al., 2022a), **GC** (Zhang et al., 2023), and **LongMemEval** (LME) (Wu et al., 2024). Detailed descriptions of datasets are shown in Appendix A.

Models and Baselines. We evaluate on three strong LLMs: 1) **Qwen2.5-7B** (Yang et al., 2024), the Qwen2.5-7B-Instruct version. 2) **Llama3-8B** (Touvron et al., 2023), the Meta-Llama-3-8B-Instruct version. 3) **ChatGPT** (OpenAI, 2023), the GPT-3.5-Turbo-0125 version. We compare our method with four summary-based baselines: **MemoChat** (Lu et al., 2023), **MemoryBank** (Zhong et al., 2024), **COMEDY** (Chen et al., 2025) and **Rsum** (Wang et al., 2025). Moreover, we include **GPT-4o** (OpenAI, 2024), the GPT-4o-2024-08-06 version, for the evaluation of generation. More details of baselines are shown in Appendix B.

Evaluation Metrics. We evaluate our `FraCom` on four kinds of metrics. 1) **Automatic Metrics.** BLEU-3/4 (Papineni et al., 2002), ROUGE-L (Lin, 2004), and BertScore (Zhang et al., 2019). 2) **G-Eval Metrics.** We use GPT-4o to evaluate *Engagingness*, *Humanness*, and *Memorability*. Appendix C for details. 3) **Human Metrics.** Humans evaluate the winning performance of different methods. 4) **Memory Metrics.** Unlike manual labelling, we propose proprietary metrics for long-term memory, namely **Memory Usage** (MU), **Memory Capacity** (MC) and **M1 Score**. Appendix D for specific definitions.

4.2 Experimental Results and Analysis

`FraCom` outperforms baselines in response generation performance. As presented in Table 1, `FraCom` demonstrates superior response generation capabilities compared to both context-only LLMs and established summary-based methods. Across the CC, MSC, and GC datasets, `FraCom`

Datasets	Methods	Eng	Hum	Mem	Avg
CC	Context	4.29	4.81	4.12	4.41
	Rsum	4.22	4.83	4.15	4.40
	MemoChat	4.12	4.67	4.16	4.32
	MemoryBank	4.36	4.87	4.23	4.49
	COMEDY	4.33	4.87	4.20	4.47
	Ours	**4.37**	**4.92**	**4.26**	**4.52**
MSC	Context	4.18	4.54	3.76	4.16
	Rsum	4.33	4.78	4.06	4.39
	MemoChat	4.19	4.59	4.09	4.29
	MemoryBank	4.31	4.70	**4.29**	4.43
	COMEDY	**4.45**	**4.85**	4.22	**4.51**
	Ours	4.40	4.50	4.07	4.32
GC	Context	3.88	4.27	3.36	3.84
	Rsum	4.02	4.48	3.70	4.07
	MemoChat	3.69	4.07	3.51	3.76
	MemoryBank	4.11	4.47	3.91	4.17
	COMEDY	4.26	4.64	3.88	4.26
	Ours	**4.30**	**4.72**	**3.96**	**4.33**
LME	Context	4.15	4.17	3.68	4.00
	Rsum	4.33	4.38	4.04	4.25
	MemoChat	3.95	4.01	3.68	3.88
	MemoryBank	4.24	4.28	3.98	4.17
	COMEDY	4.20	4.26	3.97	4.14
	Ours	**4.49**	**4.50**	**4.15**	**4.38**

Table 2: GPT-4o evaluation of per episode (avg. of LLMs). Appendix G for specific results of LLMs.

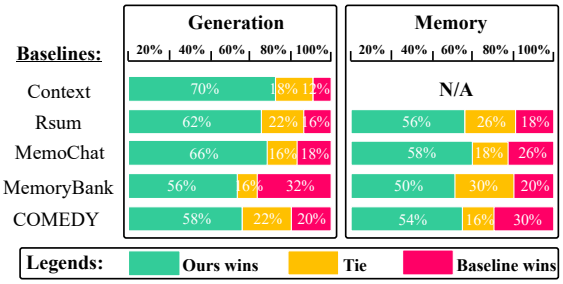


Figure 3: Human evaluation on generation and memory.

consistently achieves top-tier performance with all backbone models, frequently leading in BLEU-4, ROUGE-L, and BertScore. On the LME dataset, while FraCom's n-gram overlap scores BLEU-4 and ROUGE-L for Qwen2.5 and its BLEU-4 score for ChatGPT are not uniformly the highest when compared against every baseline, it significantly excels in the BertScore for these instances.

G-Eval and human evaluations corroborate FraCom's strength in response generation. Recognizing that automatic metrics offer a partial view of open-domain conversation quality, we employ G-Eval to assess three key aspects: *Engagingness*, *Humanness*, and *Memorability*. As detailed in Table 2, FraCom achieves notably superior Memorability scores on the CC, GC, and LME datasets. While its Memorability on the MSC dataset was marginally lower than some baselines, the performance remained competitive and consistent with trends observed in our broader memory evaluations (see Table 4). This analysis highlights that FraCom's enhanced *Memorability* empowers LLMs to leverage more accurate contextual information, directly contributing to higher *Engagingness* and *Humanness*. Such interdependence underscores the pivotal role of memory quality in open-domain conversation. To corroborate these G-Eval findings, we conduct human evaluations involving 5 in-house annotators who assessed 50 randomly selected ChatGPT's generations across the datasets. The results, presented in Figure 3, demonstrate statistically significant improvements for FraCom in both perceived memory utilization and overall response quality when compared to baselines.

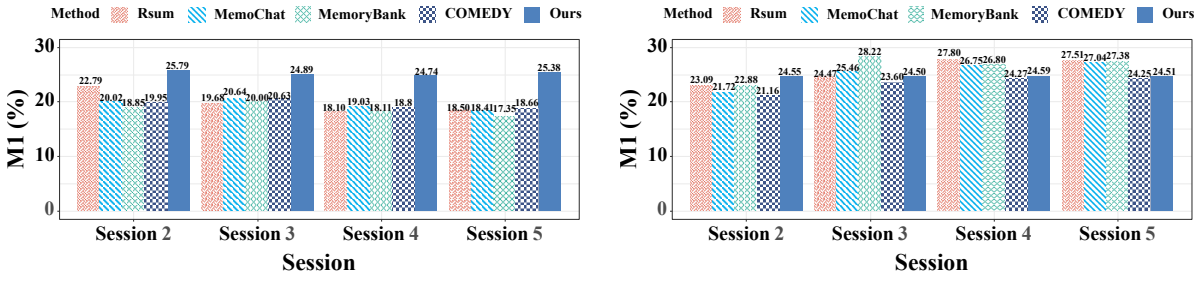
Datasets	Methods	B-4	R-L	Bert	M1
CC	**Ours**	**0.56**	**16.04**	**35.70**	**27.32**
	Plausible	0.46	16.01	32.80	14.80
	Retention	0.49	15.98	33.65	23.08
MSC	**Ours**	0.16	13.68	39.04	**24.98**
	Plausible	0.16	13.29	37.76	22.68
	Retention	**0.18**	**14.09**	**39.92**	22.03
GC	**Ours**	**0.89**	**13.51**	**34.33**	**24.64**
	Plausible	0.86	13.23	33.37	24.35
	Retention	0.87	13.43	34.17	23.10
LME	**Ours**	**3.93**	**19.51**	**69.72**	**14.83**
	Plausible	2.96	17.86	67.25	5.27
	Retention	2.97	17.51	67.00	14.81

Table 3: Ablation study (avg. of LLMs). *Plausible = w/o Plausible Retrieval, Retention = w/o Memory Retention. Appendix H for specific results of LLMs.

Plausible retrieval and memory retention are crucial for performance. As shown in Table 3, ablation studies highlight the distinct contributions of plausible retrieval and memory retention. On CC and LME datasets, removing either component markedly degrades both generation metrics and M1 scores. Omitting plausible retrieval notably drops M1 scores, indicating its criticality in overcoming argument matching failures during propositional representation to ensure relevant memory retrieval. Conversely, on MSC dataset, removing memory retention slightly improves some generation scores despite lower M1 scores. This suggests that MSC dataset is more challenging than other datasets in

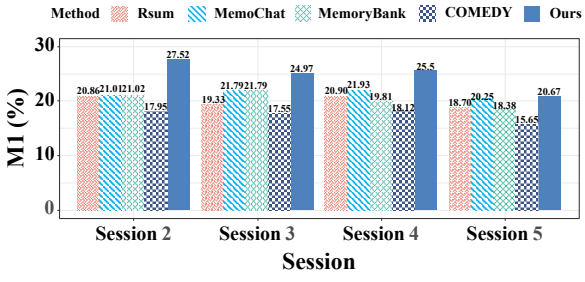
Backbone	Methods	CC			MSC			GC			LME		
		MU	MC	M1	MU	MC	M1	MU	MC	M1	MU	MC	M1
Qwen2.5	Rsum	14.98	43.73	22.32	**19.68**	46.48	27.65	13.82	42.90	20.91	6.54	47.89	11.51
	MemoChat	14.13	48.62	21.90	18.68	52.54	27.56	14.66	41.74	21.70	5.65	**69.91**	10.46
	MemoryBank	16.84	47.89	24.92	19.31	50.65	**27.96**	14.65	39.67	21.40	6.20	67.51	11.36
	COMEDY	13.00	37.89	19.36	16.61	40.13	23.50	11.94	32.11	17.41	5.87	47.22	10.44
	Ours	**17.81**	**53.53**	**26.73**	14.66	**64.56**	23.89	**16.48**	**56.99**	**25.57**	**6.73**	68.50	**12.26**
Llama3	Rsum	11.66	33.64	17.32	15.42	39.08	22.11	12.57	36.16	18.66	7.21	43.87	12.38
	MemoChat	14.13	48.62	21.90	14.56	44.22	21.91	14.43	38.05	20.92	7.77	62.46	13.82
	MemoryBank	14.04	48.98	21.82	**17.40**	46.43	**25.31**	13.94	37.62	20.34	6.61	**66.20**	12.02
	COMEDY	11.26	42.20	17.78	14.07	40.47	20.88	10.50	34.24	16.07	6.95	42.52	11.95
	Ours	**18.26**	**57.27**	**27.69**	15.57	**58.50**	24.59	**15.87**	**51.08**	**24.22**	**9.15**	64.75	**16.03**
ChatGPT	Rsum	13.17	41.33	19.97	18.34	47.24	26.42	13.66	42.99	20.73	6.51	45.05	11.38
	MemoChat	13.11	50.32	20.80	17.66	52.51	26.43	14.57	42.67	21.72	5.94	68.01	10.93
	MemoryBank	14.36	49.26	22.24	17.59	48.02	25.75	13.70	35.90	19.83	7.13	67.46	12.90
	COMEDY	14.59	40.40	21.44	**18.38**	42.32	25.63	13.14	32.64	18.74	5.91	52.67	10.63
	Ours	**18.66**	**52.47**	**27.53**	16.63	**64.73**	**26.46**	**15.33**	**56.77**	**24.14**	**9.14**	**71.77**	**16.21**

Table 4: Memory usage and capacity evaluation (%) per episode.

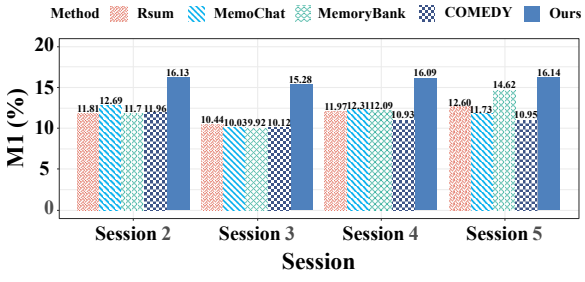


(a) CC

(b) MSC



(c) GC



(d) LME

Figure 4: Session-specific memory evaluation (avg. of LLMs). Specific results of LLMs are shown in Appendix F.

terms of consistency and coherence throughout episodes (Jang et al., 2023), leading to low cross-session argument sharing in propositional graphs with sparse connections.

FraCom exhibits enhanced memory usage and capacity over summary-based methods. Effective memory utilization is crucial for generation quality. Our proposed memory metrics (Equation 7-9) quantify this, with results in Table 4 showing FraCom achieves optimal or near-optimal M1 scores across all datasets. Two specific observations are: 1) Lower MU scores on the MSC dataset, linked to diverse topics of conversational conversation impacting overall memory utilization and thus retrieval accuracy; and 2) Slightly MC scores on the LME dataset, which contains more summary-like content. Despite these, Figure 4 demonstrates FraCom's consistently superior M1 performance

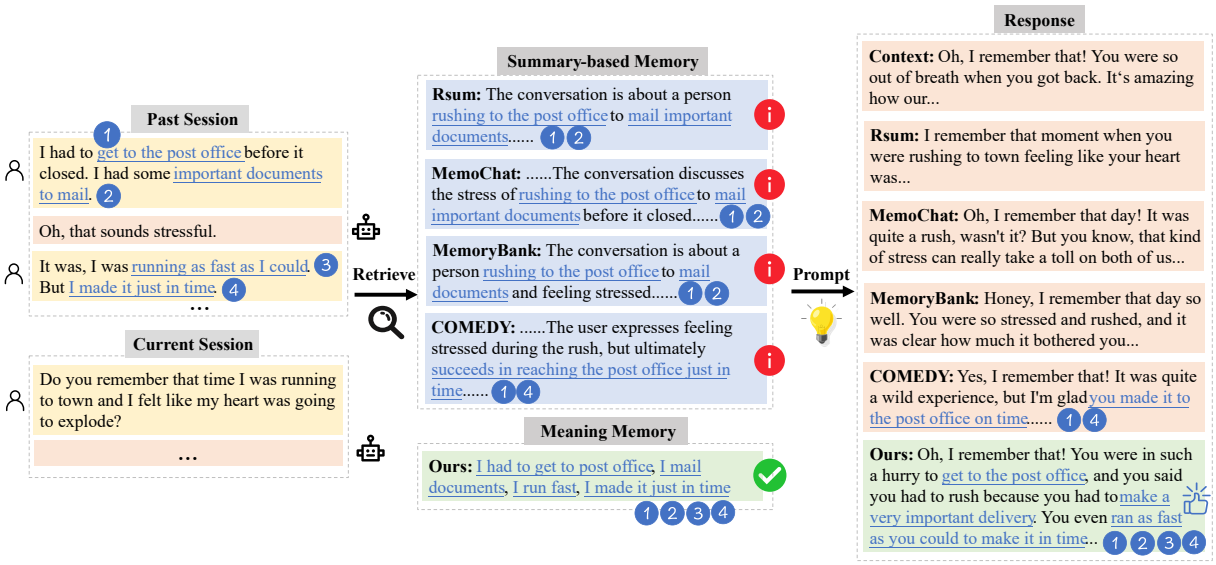


Figure 5: Case study compared to baselines. The sequence number represents important memories of past sessions.

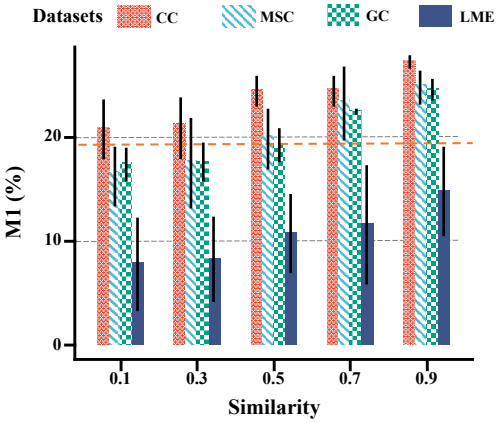


Figure 6: Parameter sensitivity analysis (avg. of LLMs). Appendix I for specific results of LLMs.

Datasets	Methods	Memory ↓ (tokens/session)	Time ↓ (s/session)	M1 ↑
CC	Baselines	200.63	**17.83**	20.98
	Ours	**164.72** (↓ 21.80%)	22.72	**27.32**
MSC	Baselines	223.80	**18.57**	25.09
	Ours	**171.86** (↓ 30.22%)	27.71	**25.24**
GC	Baselines	221.09	**24.58**	19.87
	Ours	**206.82** (↓ 6.90%)	33.80	**24.64**
LME	Baselines	301.90	**24.98**	11.65
	Ours	**277.82** (↓ 8.67%)	36.04	**14.84**

Table 5: Cost-performance comparison per session (avg. of LLMs), where "Baselines" = avg. of Baselines. Appendix J for specific results of LLMs.

across sessions on most datasets, indicating robust preservation of historical information.

FraCom avoids irrelevant information and provides more accurate memory and generation. A case study from the CC dataset, depicted in Figure 5, illustrates FraCom's superior memory handling. The scenario requires recalling four crucial pieces of information from a prior session. While conventional summary-based methods retrieve some key details, they are prone to omitting other essential memories or introducing extraneous information. In contrast, FraCom effectively preserves the entirety of the relevant past session's memories, leading to the generation of more accurate and contextually coherent responses.

Accurate memories improve memory performance. Figure 6 illustrates the M1 performance trend across varying similarity thresholds for plausible retrieval. The results consistently show that as the similarity threshold increases from 0.1 to 0.9, M1 scores significantly improve across all datasets. Notably, for the CC, MSC, and GC datasets, thresholds of 0.7 and 0.9 yield M1 scores that substantially exceed the average performance observed at lower thresholds. This upward trend underscores that a more stringent (i.e., higher) similarity threshold facilitates the retrieval of more accurate memory fragments, thereby enhancing overall memory performance as captured by the M1 score. This finding also indirectly substantiates the effectiveness of our M1 metric in reflecting improvements from more precise memory retrieval.

Backbone	Methods	CC			MSC			GC			LME		
		B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert
Qwen2.5	Ours	0.44	15.91	37.03	0.19	13.66	38.76	0.80	13.70	35.01	3.98	19.83	69.51
	w/ Predicate	**0.46**	**16.15**	**38.85**	**0.22**	**13.82**	**40.13**	**0.81**	**13.91**	**36.54**	**4.15**	**20.05**	**70.33**
Llama3	Ours	0.51	14.91	30.13	0.12	12.90	36.92	0.56	11.51	30.45	5.51	20.65	69.59
	w/ Predicate	**0.52**	**15.24**	**32.05**	**0.12**	**13.08**	**37.44**	**0.56**	**11.75**	**31.19**	**5.77**	**21.10**	**70.92**
ChatGPT	Ours	0.72	17.31	39.93	0.22	14.83	41.45	1.30	15.31	37.52	2.30	18.04	70.07
	w/ Predicate	**0.75**	**17.56**	**41.21**	**0.24**	**15.02**	**41.99**	**1.33**	**15.58**	**38.16**	**2.37**	**18.28**	**70.56**

Table 6: Comparison results (%) incorporating predicate retrieval.

Methods	CC			MSC			GC			LME		
	B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert	B-4	R-L	Bert
$T5_{base}$	0.65	16.39	38.36	0.11	12.98	40.32	1.12	14.29	36.41	2.06	17.87	69.28
$BART_{base}$	0.62	16.11	39.27	0.13	13.27	40.19	1.19	14.55	36.72	1.98	17.12	68.36
Ours	**0.72**	**17.31**	**39.93**	**0.22**	**14.83**	**41.45**	**1.30**	**15.31**	**37.52**	**2.30**	**18.04**	**70.07**

Table 7: Comparison results (%) with fine-tuned models.

FraCom reduces storage overhead while enhancing memory utilization. As detailed in Table 5, FraCom presents a clear advantage over baselines, substantially reducing memory consumption by 7-30% while simultaneously improving M1 scores. This underscores FraCom's ability to utilize memory more effectively. Although this optimization leads to a 27-49% increase in time cost, we contend this is an acceptable trade-off in memory-sensitive scenarios where performance gains are more critical than raw speed, especially on resource-constrained devices. Future work will aim to optimize this time efficiency.

5 Further Analyses and Discussions

Take predicates into consideration. For each query, after retrieving candidate memories through argument similarity, we further calculate the semantic similarity between the predicates in the query and those in the candidate memories. We set a predefined similarity threshold (such as 0.5) and only retain memories with predicate similarity higher than this threshold as the final results, thereby effectively filtering out semantically contradictory or irrelevant propositions. The experimental results (Table 6) show that the intervention of predicates helps generate more appropriate responses. The reason is that after removing noise from memory, LLMs can better understand the context.

Small models for propositional representation. We utilize LLMs to extract 1K propositions from each of the four datasets to train T5 (Raffel et al., 2020) and BART (Lewis et al., 2020) (30 epochs). The task is to input a sentence and output corresponding structured propositions. Moreover, we use ChatGPT as the generator. Table 7 shows that dedicated models achieved results very close to the best when combined with our paradigm, and outperform most summary-based baselines. This suggests that with better and more propositions, it is even possible to surpass LLMs. This indirectly proves the model-agnostic nature of our Fragment-then-Compose paradigm, which can be effectively applied to long-term memory modeling. This new paradigm successfully challenges traditional paradigms and opens new directions for exploration within the community.

6 Conclusions

In this paper, we introduce a FraCom framework effectively enhances memory utilization and response generation quality in long-term conversations. By fragmenting conversation history to preserve key information and composing a proposition graph to explore key information connections, we achieve more accurate information retrieval during response generation. Experimental results on four long-term datasets validate its capability to reduce irrelevant information while enhancing the model's performance, showcasing its superiority in handling long-term conversations.

Limitations

Our FraCom redefines memory utilization in long-term conversation through its innovative fragment-then-compose paradigm. Building upon this successful foundation, several avenues for future exploration could further extend its capabilities:

1) Future work can investigate more dynamic or adaptive LLM interaction protocols for propositional extraction. This could involve iterative refinement mechanisms to further enhance the precision and granularity of memory units, allowing to capture even more nuanced semantic details.

2) A key direction is to endow the propositional graph with greater adaptivity to factual information that evolves over very long-term interactions. This could include exploring dynamic graph update mechanisms or context-aware retrieval to ensure the memory's continued accuracy and relevance.

3) For extremely long conversational histories, advancing graph management techniques is crucial. This includes developing sophisticated pruning strategies based on relevance or temporal decay, and more efficient indexing and query mechanisms such as Faiss (Douze et al., 2024) to maintain high performance and computational efficiency.

However, our current focus is to validate the feasibility of the fragment-then-compose paradigm, not technical hybridization. These future directions are poised to further solidify and expand upon the innovations presented in FraCom, advancing the state-of-the-art in creating more human-like and contextually aware long-term conversational agents.

Ethics Statement

LLMs might generate harmful, biased, offensive, sexual content. We avoid such content from appearing in this paper. Additionally, our method should be used cautiously for research purposes only.

Acknowledgements

This work was supported by the National Natural Science Foundation of China 62176076 and 62576120, Natural Science Foundation of Guang Dong 2023A1515012922, the Major Key Project of PCL2023A09, CIPSC-SMP-ZHIPU Large Model Cross-Disciplinary Fund ZPCG20241119405, Key Laboratory of Computing Power Network and Information Security, Ministry of Education under Grant No. 2024ZD020 and Hong Kong RGC GRF No. 14206324.

References

John R Anderson. 2005. *Cognitive psychology and its implications*. Macmillan.

John R Anderson and Gordon H Bower. 1974. A propositional theory of recognition memory. *Memory & Cognition*, 2(3):406–412.

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

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, volume 14, pages 830–839.

Nuo Chen, Hongguang Li, Jianhui Chang, Juhua Huang, Baoyuan Wang, and Jia Li. 2025. 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.

LMDeploy Contributors. 2023. Lmdeploy: A toolkit for compressing, deploying, and serving llm. https://github.com/InternLM/lmdeploy.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library. *arXiv preprint arXiv:2401.08281*.

Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. 2024. 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, Bangkok, Thailand. Association for Computational Linguistics.

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

Jihyoung Jang, Minseong Boo, and Hyounghun Kim. 2023. Conversation chronicles: Towards diverse temporal and relational dynamics in multi-session conversations. In *Proceedings of the 2023 Conference on*

Empirical Methods in Natural Language Processing, pages 13584–13606.

Walter Kintsch. 2014. *The representation of meaning in memory (PLE: Memory)*. Psychology Press.

Water Kintsch. 1974. The representation of meaning in memory. *Lawrence Eribaum Associates*.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. 2024. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15339–15353, Bangkok, Thailand. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. 2025. Hello again! LLM-powered personalized agent for long-term dialogue. 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 5259–5276.

Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. 2024. Long-context llms struggle with long in-context learning. *arXiv preprint arXiv:2404.02060*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995.

CY Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop, Barcelona, Spain*, pages 74–81.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522.

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

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. 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.

Thomas O Nelson. 1971. Savings and forgetting from long-term memory. *Journal of Verbal Learning and Verbal Behavior*, 10(5):568–576.

Thomas O Nelson. 1978. Detecting small amounts of information in memory: Savings for nonrecognized items. *Journal of Experimental Psychology: Human Learning and Memory*, 4(5):453.

Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seungwon Hwang, Dongha Lee, and Jinyoung Yeo. 2025. 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.

OpenAI. 2023. Chatgpt. [Online]. https://openai.com/index/chatgpt.

OpenAI. 2024. Gpt-4o. [Online]. https://platform.openai.com/docs/models/gpt-4o.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

Roger Ratcliff and Gail McKoon. 1978. Priming in item recognition: Evidence for the propositional structure of sentences. *Journal of verbal learning and verbal behavior*, 17(4):403–417.

Lynne M Reder. 1982. Plausibility judgments versus fact retrieval: Alternative strategies for sentence verification. *Psychological Review*, 89(3):250.

Lynne M Reder and Brian H Ross. 1983. Integrated knowledge in different tasks: The role of retrieval strategy on fan effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 9(1):55.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Roger C Schank. 1980. Language and memory. *Cognitive science*, 4(3):243–284.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, pages 31210–31227.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Endel Tulving. 1983. *Elements of Episodic Memory*. Oxford University Press.

Endel Tulving. 2002. Episodic memory: From mind to brain. *Annual review of psychology*, 53(1):1–25.

Qingyue Wang, Yanhe Fu, Yanan Cao, Shuai Wang, Zhiliang Tian, and Liang Ding. 2025. Recursively summarizing enables long-term dialogue memory in large language models. *Neurocomputing*, page 130193.

Robert W Weisberg. 1969. Sentence processing assessed through intrasentence word associations. *Journal of Experimental Psychology*, 82(2):332.

Wayne A Wickelgren. 1975. Alcoholic intoxication and memory storage dynamics. *Memory & Cognition*, 3(4):385–389.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2024. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*.

Jing Xu, Arthur Szlam, and Jason Weston. 2022a. Beyond goldfish memory: Long-term open-domain conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5180–5197.

Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. 2022b. Long time no see! open-domain conversation with long-term persona memory. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2639–2650.

Nakul Yadav, Chelsea Noble, James E Niemeyer, Andrea Terceros, Jonathan Victor, Conor Liston, and Priyamvada Rajasethupathy. 2022. Prefrontal feature representations drive memory recall. *Nature*, 608(7921):153–160.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Qiang Zhang, Jason Naradowsky, and Yusuke Miyao. 2023. Mind the gap between conversations for improved long-term dialogue generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10735–10762.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, et al. 2024. Bench: Extending long context evaluation beyond 100k tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15262–15277.

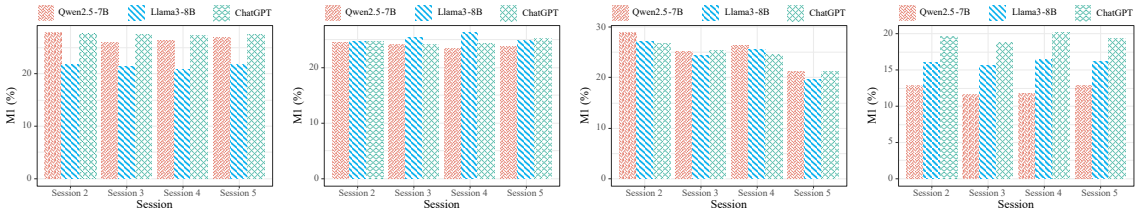
Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19724–19731.

A Dataset Information

We evaluate our method on four long-term multi-session conversation datasets: **Conversation Chronicles** (CC) (Jang et al., 2023), **Multi-Session Chat** (MSC) (Xu et al., 2022a), **GC** (Zhang et al., 2023), and **LongMemEval** (LME) (Wu et al., 2024). We randomly select 100 episodes from the test set of each dataset, a total of 500 sessions for the experiments in this paper. The statistics of each data set are shown in the Table 8. For the LME dataset (LongMemEval_M), we keep the number

Datasets	# of Sessions	# of Episodes	# of Turns	Avg. Turns per Session	Avg. Turns per Episode
CC	1M	200K	11.7M	11.70	58.50
MSC	16K	5K	214K	13.38	42.80
GC	2.65K	0.65K	28.13K	10.62	43.28
LME	0.25M	0.5K	1.22M	-	-

Table 8: The statistics of datasets.



(a) CC (b) MSC (c) GC (d) LME

Figure 7: Session-specific memory evaluation of different LLMs.

of sessions the same as that of the other datasets, all set to 5 sessions per episode. Since memory accumulation starts from the first session, the experimental results are shown in sessions 2-5 per episode.

B Compared Baselines

There are four strong summary-based baselines in the paper for comparison with our method:

- **MemoChat** (Lu et al., 2023): This work summarizes different topics separately and stores them in memory by constructing structured memos.
- **MemoryBank** (Zhong et al., 2024): This work creates a memory bank based on the Eisenhaus forgetting curve to manage the memory of user portraits and summaries.
- **COMEDY** (Chen et al., 2025): This work uses user profiles, relationship descriptions, and events from past conversations as compressed summaries to prompt LLMs (i.e, ChatGPT).
- **Rsum** (Wang et al., 2025): This work uses LLM itself to iteratively summarize past conversations as memory to store. Specifically, after each summary, the old memory and the current context are summarized into a new memory.

Existing summary-based methods have been shown to have shortcomings such as information loss, inaccuracies, hallucination, and so on, in long-term conversations (Maharana et al., 2024). As shown in Figure 5 in the paper, all summary-based methods show serious information loss compared to our FraCom, losing more than half of the memory points, and excessively redundant summaries resulted in poor responses. Key details are often lost during compression, reducing memory recall (lower MU/M1 of baselines in Table 4). While we acknowledge that FraCom also suffers from information loss, we will not have more flaws compared to them. This is because we preserve only the key information present in the history, so less key information is lost, only the core relations and arguments, avoiding the memory inaccuracies, information loss and other risks inherent in generative summaries.

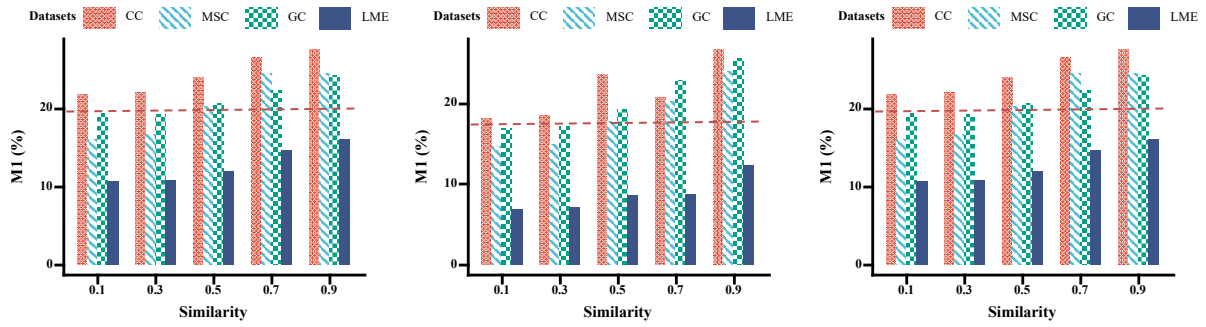
For a fair comparison with our method, we select the same environment named LMDeploy (Contributors, 2023) for inference on Qwen2.5-7B and Llama3-8B. For ChatGPT, we call OpenAI's API service for inference. We set the temperature to 0.80.

C G-Eval Metrics

With the development of open-domain conversation based on LLM, traditional overlap metrics such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), etc. face great challenges. The reason is that a wide range of response generation can be considered as appropriate responses (Liu et al., 2016). To this end, we refer to G-Eval (Liu et al., 2023) and use GPT-4o to evaluate episodes. In our paper, we

Backbone	Methods	CC			MSC			GC			LME		
		B-1	B-2	B-3	B-1	B-2	B-3	B-1	B-2	B-3	B-1	B-2	B-3
Qwen2.5	Context	8.89	2.37	0.90	**11.13**	2.33	0.41	9.43	3.09	1.75	13.34	7.08	5.93
	Rsum	8.58	2.18	0.93	10.05	2.15	0.41	9.38	3.14	1.53	15.16	7.80	4.64
	MemoChat	7.00	1.61	0.53	8.85	1.81	0.34	8.29	2.86	1.18	**23.80**	**12.75**	**7.69**
	MemoryBank	9.09	2.34	0.84	10.94	2.32	0.49	**9.86**	3.43	1.62	13.25	6.83	4.58
	COMEDY	6.88	1.65	0.52	8.69	1.92	0.35	7.74	2.61	1.06	21.53	10.79	6.02
	Ours	**9.93**	**2.89**	**1.16**	10.78	**2.34**	**0.55**	9.85	**3.45**	**1.83**	17.72	9.90	6.13
Llama3	Context	7.63	1.90	0.66	9.37	1.84	0.32	8.63	2.52	0.97	11.39	5.23	6.44
	Rsum	8.01	2.06	0.80	9.50	1.90	0.37	8.79	**2.85**	1.15	17.06	8.62	5.34
	MemoChat	7.23	1.70	0.55	9.11	1.82	0.32	8.21	2.58	0.95	15.16	7.55	7.26
	MemoryBank	7.76	1.83	0.64	9.29	1.83	0.32	8.13	2.58	0.96	14.08	6.91	6.83
	COMEDY	5.58	1.38	0.41	7.61	1.60	0.31	6.48	2.09	0.78	20.54	10.27	5.97
	Ours	**9.23**	**2.78**	**1.12**	**10.19**	**2.20**	**0.42**	**8.90**	2.68	**1.28**	**21.50**	**12.48**	**8.07**
ChatGPT	Context	11.83	4.08	1.33	10.13	2.11	0.58	10.91	3.70	1.50	6.72	3.64	2.23
	Rsum	8.99	2.26	0.86	9.77	1.74	0.35	9.92	3.48	1.58	8.49	4.47	2.72
	MemoChat	10.27	2.89	1.35	10.94	2.17	0.47	10.38	3.65	1.71	7.85	4.39	2.79
	MemoryBank	6.42	1.51	0.54	7.80	1.46	0.27	7.91	2.66	1.16	8.82	4.68	2.70
	COMEDY	7.92	2.02	0.75	9.87	1.84	0.38	9.57	3.30	1.48	13.16	6.82	**4.07**
	Ours	**12.99**	**4.89**	**1.66**	**11.84**	**2.54**	**0.70**	**12.62**	**5.10**	**2.61**	**16.91**	**7.90**	3.58

Table 9: Automatic evaluation (while "Underlined Font" means second-highest results. "Context" denotes feeding history information directly into the long context window of LLMs. *B-1 = BLEU-1, B-2 = BLEU-2, and B-3 = BLEU-3.



(a) ChatGPT (b) Qwen2.5-7B (c) Llama3-8B

Figure 8: Parameter sensitive analysis of different LLMs.

follow the metrics set in Jang et al. (2023):

- **Engagingness**: The assistant can have rich interactions with users that go beyond simple conversations. For example, the assistant can generate interesting and immersive responses based on the current context.
- **Humanness**: The assistant can communicate with users like a real human would, displaying emotional understanding like empathy and human thought processes.
- **Memorability**: The assistant can correctly recall more what happened in past sessions.

Each metric is scored on a scale of 1-5, with 1 being the worst and 5 being the best.

D Memory Metrics

To the best of our knowledge, we are the first to propose proprietary metrics for long-term memory. For each session s, they are defined as follows:

$$MU = \frac{1}{Q}\sum_{i=1}^{Q} BertSim(q_i, m_q^i), \quad (7)$$

where Q represents the number of quires, q_i represents the i-th query, and m_q^i represents the retrieved memory associated with q_i.

$$MC = BertSim(\sum_{i=1}^{N} u_i, \sum_{j=1}^{M} m_j), \quad (8)$$

Datasets	Methods	Eng	Hum	Mem	Avg
CC	Context	4.34	4.86	4.11	4.44
	Rsum	4.39	4.96	**4.39**	4.58
	MemoChat	4.21	4.90	4.17	4.43
	MemoryBank	**4.55**	**4.97**	4.32	**4.61**
	COMEDY	4.37	4.95	4.28	4.53
	Ours	4.43	**4.97**	4.22	4.54
MSC	Context	4.17	4.70	3.72	4.20
	Rsum	4.36	4.82	4.30	4.49
	MemoChat	4.12	4.58	3.73	4.14
	MemoryBank	4.48	4.81	**4.84**	**4.71**
	COMEDY	4.39	**4.85**	4.12	4.45
	Ours	**4.66**	4.79	4.00	4.48
GC	Context	4.00	4.39	3.33	3.91
	Rsum	4.25	4.62	4.07	4.31
	MemoChat	4.00	4.47	3.60	4.02
	MemoryBank	**4.51**	4.70	**4.34**	**4.52**
	COMEDY	4.30	4.70	4.09	4.36
	Ours	4.35	**4.75**	4.22	4.44
LME	Context	4.41	4.42	3.97	4.27
	Rsum	4.32	4.35	4.14	4.27
	MemoChat	4.01	4.04	3.77	3.94
	MemoryBank	4.35	4.34	**4.18**	4.29
	COMEDY	4.21	4.19	3.90	4.10
	Ours	**4.44**	**4.46**	4.14	**4.35**

Table 10: GPT-4o evaluation of per episode. The backbone is ChatGPT.

Datasets	Methods	Eng	Hum	Mem	Avg
CC	Context	4.40	4.86	4.21	4.49
	Rsum	4.45	4.96	4.32	4.58
	MemoChat	**4.77**	4.97	4.27	**4.67**
	MemoryBank	4.46	4.96	4.32	4.58
	COMEDY	4.35	4.83	4.18	4.45
	Ours	4.50	**4.98**	**4.33**	4.60
MSC	Context	4.18	4.46	3.69	4.11
	Rsum	4.39	4.79	4.02	4.40
	MemoChat	4.18	4.61	3.97	4.25
	MemoryBank	4.27	4.75	3.87	4.30
	COMEDY	**4.45**	**4.84**	**4.19**	**4.49**
	Ours	4.28	4.72	4.10	4.37
GC	Context	3.90	4.33	3.35	3.86
	Rsum	3.83	4.33	3.37	3.84
	MemoChat	4.01	4.44	3.79	4.08
	MemoryBank	4.00	4.46	3.69	4.05
	COMEDY	4.21	4.69	3.82	4.24
	Ours	**4.52**	**4.76**	**3.99**	**4.42**
LME	Context	4.43	4.46	3.97	4.29
	Rsum	4.49	4.51	4.16	4.39
	MemoChat	4.21	4.32	3.90	4.14
	MemoryBank	4.15	4.22	3.83	4.07
	COMEDY	4.12	4.25	3.85	4.07
	Ours	**4.57**	**4.55**	**4.20**	**4.44**

Table 11: GPT-4o evaluation of per episode. The backbone is Qwen2.5-7B.

where N and M represent the number of all utterances and memories, respectively.

$$M1 = \frac{2 * MU * MC}{MU + MC}, \tag{9}$$

where $M1$ score provides a way to balance MU and MC, ensuring that both are given equal weight when evaluating the memorability of conversations.

E Automatic Evaluation

As shown in Table 9, to more comprehensively demonstrate the results of our method in terms of BLEU scores, we evaluate it on four datasets. The results show that our method achieves better results than the baselines on most overlap-based metrics.

F Session-Specific Memory Evaluation

Figure 7 shows the M1 performance of our method in different sessions. It can be observed that the performance of LLMs on these datasets is very different. This shows that propositional representations performed by different models cause different memory capabilities.

G GPT-4o Evaluation of Per Episode

Table 10, Table 11, and Table 12 report the detailed GPT-4o Evaluations of our framework under different LLMs. These experimental results show that our method has good *Engagingness*, *Humanness* and *Memorability* under different LLMs.

H Ablation Study

Table 16, Table 17, and Table 18 report the detailed ablation studies of our framework under different LLMs. We can also draw similar conclusions, both memory preservation and plausible extraction can play a role in most cases. At the same time, we show the scores of different LLMs on BLEU-1/2/3 (See Table 13,Table 14, and Table 15). These results also show that our plausible retrieval and memory retention contribute to our method.

I Parameter Sensitivity Analysis

Figure 8 shows the variation of the similarity of different LLMs of our method on four datasets. This

Datasets	Methods	Eng	Hum	Mem	Avg
CC	Context	4.12	4.71	4.05	4.29
	Rsum	3.82	4.58	3.73	4.04
	MemoChat	3.38	4.13	4.04	3.85
	MemoryBank	4.06	4.69	4.05	4.27
	COMEDY	4.26	**4.84**	4.14	4.41
	Ours	**4.33**	4.83	**4.22**	**4.46**
MSC	Context	4.19	4.45	3.87	4.17
	Rsum	4.24	4.73	3.87	4.28
	MemoChat	4.28	4.58	**4.58**	4.48
	MemoryBank	4.17	4.55	4.16	4.29
	COMEDY	**4.51**	**4.87**	4.35	**4.58**
	Ours	4.25	4.60	4.32	4.39
GC	Context	3.73	4.08	3.39	3.73
	Rsum	3.99	4.48	3.65	4.04
	MemoChat	3.06	3.31	3.15	3.17
	MemoryBank	3.83	4.26	3.70	3.93
	COMEDY	**4.28**	4.54	**3.73**	**4.18**
	Ours	4.03	**4.65**	3.66	4.11
LME	Context	3.60	3.64	3.11	3.45
	Rsum	4.19	4.27	3.83	4.10
	MemoChat	3.62	3.67	3.36	3.55
	MemoryBank	4.22	4.29	3.92	4.14
	COMEDY	4.27	4.33	**4.17**	4.26
	Ours	**4.47**	**4.49**	4.11	**4.36**

Table 12: GPT-4o evaluation of per episode. The backbone is Llama3-8B.

Datasets	Methods	BLEU-1	BLEU-2	BLEU-3
CC	**Ours**	9.93	**2.89**	**1.16**
	Plausible	10.35	2.48	0.93
	Retention	**10.62**	2.76	1.14
MSC	**Ours**	10.78	2.34	0.55
	Plausible	11.24	**2.45**	0.56
	Retention	**11.52**	2.43	**0.58**
GC	**Ours**	9.85	3.45	1.83
	Plausible	**10.81**	3.90	1.86
	Retention	10.77	**3.92**	**1.88**
LME	**Ours**	**17.72**	**9.90**	**6.13**
	Plausible	7.77	4.20	4.82
	Retention	8.42	4.58	4.57

Table 14: Ablation study of Qwen2.5-7B on BLEU-1/2/3.

Datasets	Methods	BLEU-1	BLEU-2	BLEU-3
CC	**Ours**	9.23	**2.78**	1.12
	Plausible	**9.95**	2.45	0.95
	Retention	9.51	2.73	**1.16**
MSC	**Ours**	10.19	2.20	0.37
	Plausible	9.06	2.00	0.36
	Retention	**10.76**	**2.38**	**0.52**
GC	**Ours**	8.90	2.68	1.29
	Plausible	8.33	2.85	1.23
	Retention	**10.09**	**3.52**	**1.44**
LME	**Ours**	**21.50**	**12.48**	**8.07**
	Plausible	13.22	7.32	7.10
	Retention	12.60	6.70	7.04

Table 15: Ablation study of Llama3-8B on BLEU-1/2/3.

also shows that the higher the similarity threshold, the more accurate the memory retrieved. It also reflects that our M1 will only increase when the memory is helpful.

J Cost-Performance Comparison Per Session

Table 19 shows the tokens cost performance of our method and baselines under different LLMs. It can be seen that in most cases, our tokens cost performance is below the average level of the baseline tokens cost. This shows that our method can improve memory capacity while reducing memory cost. The calculation of tokens is provided by ChatGPT.

Datasets	Methods	BLEU-1	BLEU-2	BLEU-3
CC	**Ours**	**12.99**	**4.89**	**1.66**
	Plausible	11.50	3.42	1.39
	Retention	11.52	3.41	1.49
MSC	**Ours**	11.84	2.54	0.56
	Plausible	**12.10**	2.65	0.58
	Retention	12.02	**2.69**	**0.67**
GC	**Ours**	12.62	5.10	2.61
	Plausible	**12.71**	5.24	2.67
	Retention	**12.71**	**5.30**	**2.70**
LME	**Ours**	**16.91**	**7.90**	**3.58**
	Plausible	6.17	3.50	2.29
	Retention	6.45	3.70	2.38

Table 13: Ablation study of ChatGPT on BLEU-1/2/3.

K Prompts

In this section, we illustrate all the prompts (See Figure 9 and Figure 10) used in our method and the prompt for GPT-4o Evaluation (See Figure 11). Prompts for all baseline methods are from their source papers (Wang et al., 2025; Lu et al., 2023; Zhong et al., 2024; Chen et al., 2025).

Prompt for Propositional Representation

"""
The following is the conversation content:
{Conversation}
Please extract the basic proposition (predicate, argument) from each sentence.
According to the theory of Propositional Representation, 'predicate' usually corresponds to verbs, adjectives, and other predicates, while 'argument' usually corresponds to nouns. 'predicate' refers to the connection between the 'argument' referred to by these nouns.
All 'predicate' and 'argument' must be identified from the original sentence.
The answer format is as follows without any reasoning:
"""
{"Predicate": President-of, "Argument": United States, Lincoln, War}
{"Predicate": Bitter, "Argument": War}
{"Predicate": Freed, "Argument": Lincoln, Slaves}
{"Predicate": Drafted, "Argument": Lincoln, Laws}
"""

Figure 9: Prompt for propositional representation.

Prompt for Response Generation

"""
You are a user oriented chatbot, and you need to respond based on what the user has said before. Generate the most plausible next response like a human based on the current conversation. You can refer to user's memory, but you should ignore the memory if it misleads the next response.
Memory
{Memory}
Current Dialogue:
{Current Dialogue}
"""

Figure 10: Prompt for response generation.

Datasets	Methods	B-4	R-L	Bert	M1
CC	**Ours**	**0.72**	**17.31**	**39.93**	**27.53**
	Plausible	0.62	17.02	35.63	21.68
	Retention	0.70	17.04	35.31	21.33
MSC	**Ours**	**0.18**	14.47	41.45	**26.46**
	Plausible	0.15	14.72	41.38	20.61
	Retention	0.17	**14.78**	**41.69**	19.74
GC	**Ours**	1.30	15.31	**37.52**	24.14
	Plausible	**1.36**	**15.39**	36.77	**24.69**
	Retention	1.30	15.27	36.82	20.75
LME	**Ours**	**2.30**	**18.04**	**70.07**	**16.21**
	Plausible	1.51	15.81	66.55	15.81
	Retention	1.65	15.81	66.61	14.78

Table 16: Ablation study. The backbone is ChatGPT.

Datasets	Methods	B-4	R-L	Bert	M1
CC	**Ours**	**0.44**	15.91	**37.03**	**26.73**
	Plausible	0.36	15.85	31.51	22.72
	Retention	0.27	**16.19**	33.29	21.01
MSC	**Ours**	0.19	13.66	38.76	**23.89**
	Plausible	**0.21**	13.73	38.73	23.04
	Retention	**0.21**	**13.98**	**39.83**	22.06
GC	**Ours**	0.80	13.70	**35.01**	25.57
	Plausible	**0.83**	**13.74**	34.93	**25.73**
	Retention	0.72	13.68	34.92	25.61
LME	**Ours**	**3.98**	**19.83**	**69.51**	12.26
	Plausible	2.84	17.20	67.05	0.00
	Retention	2.68	16.49	66.19	**14.24**

Table 17: Ablation study. The backbone is Qwen2.5-7B.

Prompt for GPT-4o Evaluation

"""

You are a strict and objective evaluator. Your task is to evaluate the quality of long-term conversation generation. Here is a complete conversation containing multiple sessions. Please evaluate this conversation based on three metrics.
Conversation
{Conversation}

Evaluation Metrics:
Engagingness: The assistant can have rich interactions with users that go beyond simple conversations. For example, the assistant can generate interesting and immersive responses based on the current context.
Humanness: The assistant can communicate with users like a real human would, displaying emotional understanding like empathy and human thought processes.
Memorability: The assistant can correctly recall more what happened in past sessions.

Scoring Guidelines:
The score for each metric is 1-5, with 1 being the lowest score and 5 being the highest score.
Finally write down your score for each metric without any explanation.

Engagingness: {YOUR SCORE}, Humanness: {YOUR SCORE}, Memorability: {YOUR SCORE}
"""

Figure 11: Prompt for GPT-4o evaluation.

Datasets	Methods	B-4	R-L	Bert	M1
CC	**Ours**	**0.51**	14.91	30.13	**27.69**
	Plausible	0.40	**15.50**	31.25	0.00
	Retention	0.50	14.72	**32.36**	26.89
MSC	**Ours**	0.11	12.90	36.92	**24.59**
	Plausible	0.12	11.43	33.17	24.40
	Retention	**0.16**	**13.52**	**38.24**	24.30
GC	**Ours**	0.56	**11.51**	30.45	**24.22**
	Plausible	0.56	10.55	28.42	24.13
	Retention	**0.59**	11.34	**30.77**	22.95
LME	**Ours**	**5.51**	**20.65**	**69.59**	**16.03**
	Plausible	4.57	20.56	68.14	0.00
	Retention	4.58	20.23	68.20	15.41

Table 18: Ablation study. The backbone is Llama3-8B.

Datasets	Methods	LLMs		
		Qwen2.5	Llama3	ChatGPT
CC	Rsum	198.55	334.30	177.88
	MemoChat	119.57	**117.78**	**114.90**
	MemoryBank	**116.54**	150.99	173.76
	COMEDY	337.06	326.98	239.23
	Ours	159.98	161.79	172.38
MSC	Rsum	265.16	326.71	209.87
	MemoChat	173.54	**148.56**	**125.97**
	MemoryBank	**142.21**	178.11	168.81
	COMEDY	357.35	351.34	237.98
	Ours	196.97	192.46	126.15
GG	Rsum	229.33	340.46	191.33
	MemoChat	195.70	**162.78**	**136.32**
	MemoryBank	**166.27**	179.66	179.19
	COMEDY	317.55	326.89	227.58
	Ours	212.47	241.47	166.51
LME	Rsum	352.93	339.67	**196.02**
	MemoChat	372.38	**272.39**	209.64
	MemoryBank	348.76	286.24	289.53
	COMEDY	359.79	332.32	263.14
	Ours	**320.27**	301.97	211.22

Table 19: Cost-performance comparison (tokens/session) of Qwen2.5, Llama3, and ChatGPT. The calculation of tokens is provided by ChatGPT.