Evaluating the Long-Term Memory of Large Language Models

Zixi Jia and Qinghua Liu and Hexiao Li
Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China

Yuyan Chen
Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University Shanghai, China

Jiqiang Liu
Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China

Abstract

In applications such as dialogue systems, personalized recommendations, and personal assistants, large language models (LLMs) need to retain and utilize historical information over the long term to provide more accurate and consistent responses. Although long-term memory capability is crucial, recent studies have not thoroughly investigated the memory performance of large language models in long-term tasks. To address this gap, we introduce the Long-term Chronological Conversations (LOCCO) dataset and conduct a quantitative evaluation of the long-term memory capabilities of large language models. Experimental results demonstrate that large language models can retain past interaction information to a certain extent, but their memory decays over time. While rehearsal strategies can enhance memory persistence, excessive rehearsal is not an effective memory strategy for large models, unlike in smaller models. Additionally, the models exhibit memory preferences across different categories of information. Our study not only provides a new framework and dataset for evaluating the long-term memory capabilities of large language models but also offers important references for future enhancements of their memory persistence.[1]

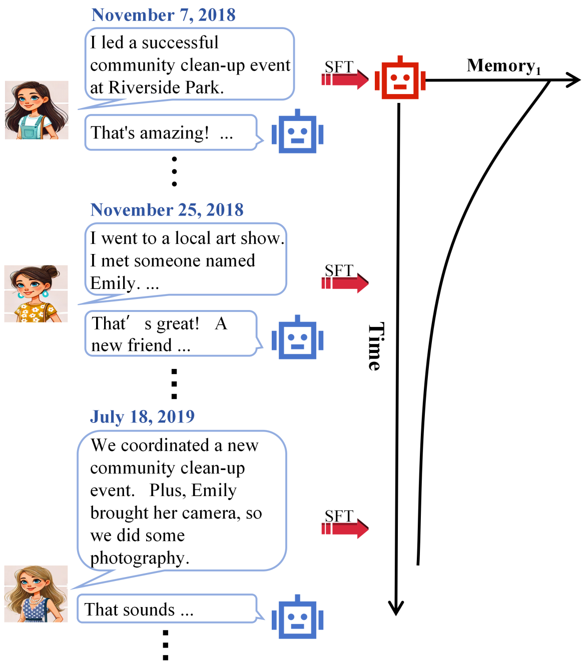

Figure 1: **An Example in LOCCO.** We impart memory to the LLMs through supervised fine-tuning and examine how this memory changes over time. Memory1 represents the model's memory of the dialogues from the first time period. The model gradually forgets the information from this initial period.

1 Introduction

In recent years, large language models (LLMs) have been widely applied across various fields, driving technological advancements. In many practical applications, such as personal assistants (Lu et al., 2023), personalized recommendations (Wang et al., 2023c), and dialogue systems (Zhong et al., 2024), models need to retain and utilize past information over the long term to provide more accurate and consistent responses. Although long-context strategies (Bertsch et al., 2024) and retrieval-augmented generation techniques (Shuster et al., 2021) have improved LLMs' memory in handling long-term tasks, these text-based memory methods face significant limitations in terms of token count, computational cost, and inference time (Zhang et al., 2024).

In contrast, parameter-based memory stores information by adjusting the model's internal parameters, meaning that this information is an inherent part of the model, better reflecting the concept of memory within the model itself. While prior work has demonstrated the memory performance of LLMs in related domains (Shao et al., 2023), their memory performance in long-term tasks remains underexplored. Considering that human-machine dialogue is a crucial application of LLMs, memory

[1]Data available at: https://github.com/JamesLLMs/LoCoGen

Findings of the Association for Computational Linguistics: ACL 2025, pages 19759–19777
July 27 - August 1, 2025

plays a key role. Evaluating LLMs' performance in long-term dialogue tasks can indirectly reflect their long-term memory capabilities (Zhang et al., 2024).

To this end, we propose a pipeline for constructing long-term dialogue data: Long Conversation Generation (LoCoGen), an automated dialogue generation pipeline based on LLMs. We use LoCoGen to build two dialogue datasets focused on evaluating LLMs' long-term memory capabilities—Long-term Chronological Conversations (LOCCO) and its extended version LOCCO-L. LOCCO contains 100 users' long-term dialogues with a chatbot, totaling 3080 interactions, simulating the application scenario of LLMs as chatbots. LOCCO-L provides even longer conversational sequences with more iterations, offering additional test cases for evaluating memory persistence across extended timelines.

Previous research has predominantly assessed memory by evaluating the extent to which models fit the training data, employing identical task formats during both the training and evaluation phases (Tirumala et al., 2022; Wang et al., 2019; Han et al., 2020). However, for LLMs, the memory process should represent an organic integration of training data, rather than mere rote memorization of its paradigms. Inspired by (Maharana et al., 2024; Du et al., 2024), we examine LLMs' memory through dialogue question-answering tasks. In our experimental setup, the model does not learn how to utilize the conversation to perform the Q&A tasks during the memory formation process. Therefore, when the model can accurately answer questions using information from the conversation, it indicates that the model has genuinely retained the conversational information. This demonstrates an organic and interactive memory process. Additionally, metrics like ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002) have limited accuracy in open-domain dialogues, so we trained a consistency model to replace existing automated metrics for assessing response accuracy.

Experiments on open-source LLMs show that they possess a certain degree of memory capability in long-term tasks, able to recall historical interaction information such as names, places, and specific events, and use this information to answer questions. However, LLMs face limitations in handling dialogues over long time spans, gradually forgetting historical dialogues. To enhance memory persistence, we employed a rehearsal strategy from continual learning. The results demonstrate that, unlike in smaller models, excessive rehearsal is not an effective memory strategy. Our contributions are as follows:

i)We provide an automated pipeline, LoCoGen, for constructing long-term dialogue data and create both the LOCCO and LOCCO-L datasets to measure LLMs' long-term memory.

ii)We quantitatively evaluate LLMs' long-term memory capabilities using LOCCO and further explore factors that may affect memory. We find that memory gradually weakens over time and that LLMs exhibit memory preferences.

iii)We found that rehearsal strategies can enhance the memory of LLMs; however, they do not prevent complete forgetting. Additionally, spaced learning is more effective than massed learning in terms of memory retention. Nevertheless, for LLMs, excessive rehearsal is not an effective memory strategy.

2 Related Works

2.1 Memory in LLMs

Previous studies have proposed several promising memory mechanisms, categorizing memory into text-based and parameter-based forms. Memory in textual form (Li et al., 2023; Huang et al., 2023; Zhong et al., 2024) offers good interpretability and implementation convenience for long-term memory in LLMs. However, it also faces challenges such as high computational cost, inference time delays, information loss, and inference robustness issues. Approaches that alter model parameters through fine-tuning (Shao et al., 2023; Wang et al., 2023b) are not constrained by the context length limitations of LLMs. They offer higher inference efficiency and lower inference costs. However, fine-tuning LLMs can lead to forgetting original knowledge due to parameter updates (Jang et al., 2021; Ke et al., 2021). This can impact the performance of LLMs on tasks requiring long-term continuous memory. Previous work has not quantitatively assessed the performance of fine-tuned memory in long-term tasks, highlighting the need for quantitative evaluation of models' memory in long-term memory tasks.

2.2 Long-term Dialogue

Recent approaches (Xu et al., 2022b; Chen et al., 2024) store memory in text form without changing model parameters, preventing models from truly remembering dialogue history. We adjust model parameters through supervised fine-tuning, enabling models to internalize key information from long-term dialogues as an inherent part. To evaluate the performance of dialogue agents in long-term conversations, some datasets have been proposed(Jang et al., 2023; Zhang et al., 2023). These datasets only cover a few to dozens of dialogue turns, lacking sufficient historical dialogue content and time span to adequately assess the long-term memory capabilities of LLMs. Maharana et al. (2024) use the F1 score as an evaluation metric for dialogue question-answering, which is insufficient to accurately assess the performance of LLMs across different formats. By introducing LoCoGen, we automatically constructed dialogue data with long-term consistency, addressing the limitations in time span and historical content of existing methods. Additionally, we provide a more precise evaluation method for long-term conversational memory.

3 Task Setup

3.1 Long-term Dialogue Memory

We denote long-term dialogue data as $D = \{D_1, D_2, ..., D_n\}$, where D_j represents the dialogue data within the T_j time period. Each D_j consists of multiple individual dialogues, i.e., $D_j = \{D_{j1}, D_{j2}, ..., D_{jm}\}$, where m is the number of dialogues within the T_j time period. We ensure that the number of dialogues in each time period is approximately equal. Q_j represents the questions posed by the user regarding the dialogues in D_j, $Q_j = \{Q_{j1}, Q_{j2}, ..., Q_{jk}\}$ (where $k \leq m$). Each question Q_{jx} uniquely corresponds to a dialogue D_{jx}. If the trained model M can accurately utilize the information in D_{jx} to answer the user's question Q_{jx}, then the model M is considered to have memory of D_{jx}.

3.2 Research Questions

We have formulated the following six research questions to explore the long-term memory capabilities of large language models: i) How do large language models perform in terms of long-term memory? ii) Does the memory performance of large language models vary with the introduction of new data? iii) Do large language models exhibit memory preferences similar to those observed in humans? iv) Do large language models experience cognitive load in a manner analogous to humans? v) Do large language models exhibit a forgetting baseline? vi)Do large language models achieve permanent memory through replay strategies comparable to those utilized by humans?

3.3 Data Construction

Long-term Chronological Conversations. Constructing long-term dialogues faces two main challenges: i) The length of text generated by LLMs is limited (e.g., GPT-4o's maximum length is 4096 tokens); ii) It is essential to ensure that the background and development trajectory of characters remain coherent throughout the dialogue, avoiding inconsistent or conflicting plots. We propose a pipeline named LoCoGen (Long Conversation Generation) that can automatically generate long and consistent dialogues based on brief character descriptions. Figure 2 shows an overview of LoCoGen.

We first selected character descriptions from the MBTI-S2Conv dataset (Tu et al., 2023) as the foundation. This dataset contains 1024 virtual characters, each with a structured data description, including name, gender, age, personality, and background. To ensure that the dialogues reflect the characters' changes, we set specific timestamps for each character description. To extend the character descriptions and simulate real-life user changes, we first used prompts to expand the initial character descriptions to cover three different time points. These time-point descriptions reflect the characters' growth and changes while maintaining consistency with their backgrounds. In this way, we initially established a timeline for each character, ensuring the rationality and consistency of character depictions across different time periods. To obtain more detailed character descriptions and showcase the characters' long-term changes in detail, we inserted new time-point descriptions between the existing time points and iterated this process. The prompts included the character descriptions from the preceding and following time points. Inspired by the plot progression techniques used by novelists in constructing long narratives, we iteratively inserted new descriptions to build more detailed long-term descriptions, ensuring the characters' development remained coherent and consistent.

After completing the long-term description of characters, we further inserted multiple events between each description to simulate the experiences of characters during that period. To ensure event consistency, we were inspired by Yang et al. (2022) and employed recursive reprompting. After gener-

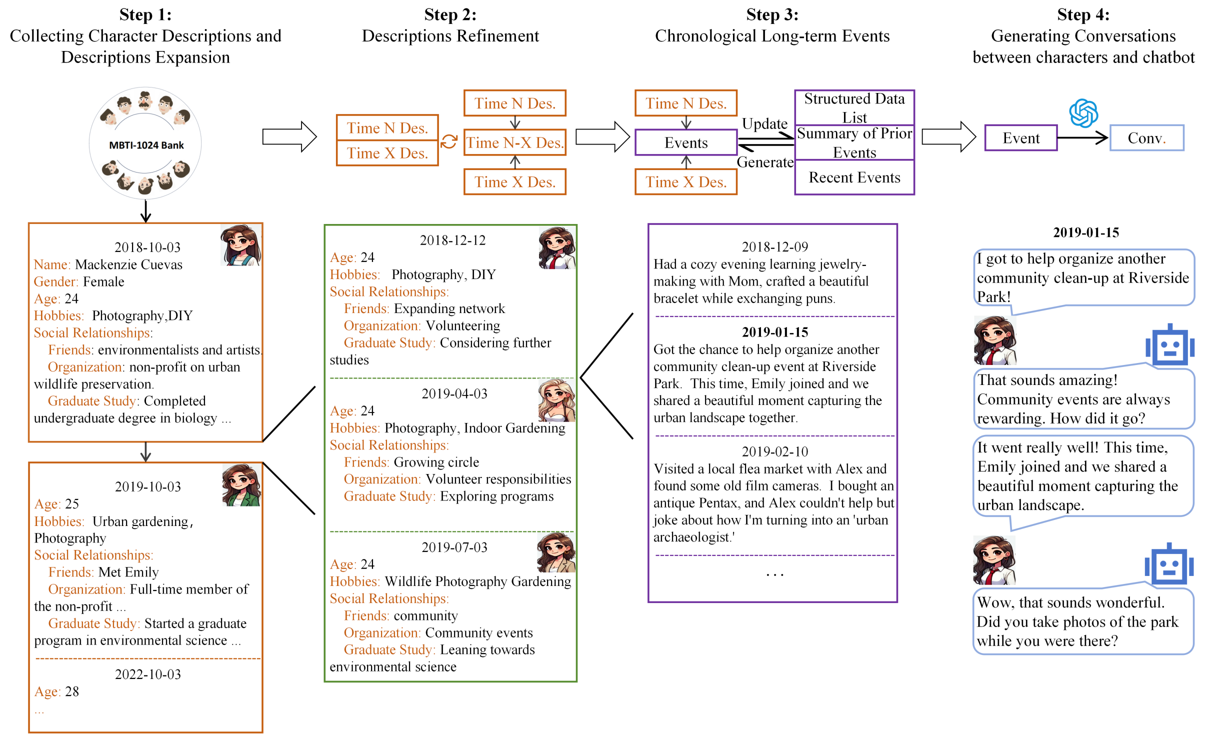

Figure 2: **Overview of LoCoGen.** We use unique character descriptions as the initialization, followed by generating a series of events and interactions related to the characters to construct the dataset. We illustrate the construction process of long-term dialogue data for a character in LOCCO, omitting some parts for brevity.

Dataset	Avg. sessions per conv.	Character Count	Avg. tokens per conv.	Time Interval	Collection
MPCChat (Ahn et al., 2023)	1	-	53.3	-	Reddit
MMDialog (Feng et al., 2022)	1	-	72.5	-	Social media
Daily Dialog (Li et al., 2017)	1	-	114.7	-	Crowdsourcing
SODA (Kim et al., 2023)	1	-	122.4	-	LLM-generated
MSC(Xu et al., 2022a) (train: 1–4 sessions)	4	-	1,225.9	few days	Crowdsourcing
Conversation Chronicles (Jang et al., 2023)	5	-	1,054.7	few hours - years	LLM-generated
LoCoMo (Maharana et al., 2024)	27.2	10	16,618.1	few months	LLM-gen.+ crowdsourc.
LOCCO (ours)	30.8	100	3,856.20	few days	LLM-generated
LOCCO-L (ours)	112.8	20	14,122.56	few days	LLM-generated

Table 1: Statistics comparing LOCCO with existing dialogue datasets, showing that the average session length of long-term dialogues in LOCCO significantly exceeds that of existing datasets.

ating each new event, we summarize past events to retain key information. Additionally, we maintain an automatically updated structured list that records information about key characters, locations, items, and other elements mentioned in the events. When generating new events, the following four components are referenced: i)Character descriptions at two time points: Ensures events align with character development; ii)Event summary: Summarizes the new event and some previous events to ensure important contextual information is retained; iii)Automatically updated structured list: This list records important elements mentioned in events (e.g., characters, locations, items) in real-time and is used to maintain consistency when generating new events; iv)Most recently generated event: Incorporates the content of the latest event into prompts to help generate subsequent events, ensuring smooth continuity with prior content. Based on long-term events, we use LLMs to generate dialogues. The generated long-term dialogues closely align with the characters' backgrounds and development trajectories. The dialogues simulate interactions between characters acting as users and the large language model. Detailed prompts used in LoCoGen can be found in Appendix A.1. We randomly selected 100 characters from the MBTI-S2Conv (Tu et al., 2023) dataset to initialize character descriptions. By running the aforementioned generation process, we constructed a long-term consistent dialogue dataset, Long-term Chronological Conversations (LOCCO), containing 3080 dialogue entries. Additionally, we created LOCCO-L by further refining the process with fewer time intervals and more iterations, using 10 characters to generate an average of 112.8 sessions per conversation.

The generated LLM data sometimes exhibit quality inconsistencies, potentially containing incorrect information or deviating from the specified format. To ensure high quality and consistency of the dataset, we implemented an automated process to filter out these issues (see detailed process in Appendix A.2). Table 1 presents the statistics of the LOCCO dataset.

We refer to (Bae et al., 2022) and employ a manual approach to evaluate the dialogue data. Specifically, we randomly selected 200 historical dialogues and required crowdworkers to rate their level of agreement with each evaluation criterion on a scale from 0 to 5. The overall results are presented in Table 2. Detailed descriptions of the evaluation criteria can be found in Appendix A.3).

Metrics	Avg	Std
Consistency	4.40	0.52
Coherence	4.45	0.78
Participation	4.58	0.86
Overall	4.47	-

Table 2: Results of Manual Evaluation of Dialogue Data.

Gao et al. (2023) has utilized LLMs as evaluators to assess data quality, demonstrating high consistency with human evaluation results. Therefore, we also use LLMs to evaluate the dialogue data, scoring dialogues in terms of Participation, Coherence, and Rationality. Detailed scoring instructions and results are provided in Appendix A.4.

Dialogue Question Answering. Considering that dialogue Q&A can effectively assess a model's memory (Maharana et al., 2024), we generated a set of dialogue Q&A pairs for each conversation, with answers intended to align with key information mentioned in the historical dialogue. The core idea of the evaluation is that if the model can accurately use key information from the historical dialogue to answer questions, it is considered to have remembered that dialogue. To ensure data quality and evaluation effectiveness, we manually filtered the Q&A pairs, ultimately retaining 2,981 dialogue Q&A pairs. For detailed construction processes and filtering rules, refer to Appendix B.

4 Experiments

4.1 Experimental Setup

We conducted experiments on 8 x NVIDIA GeForce RTX 3090 (each with 24GB) and used LLama-Factory for model training and inference, employing LoRA (Low-Rank Adaptation) for training. The training used a batch size of 1 (we found that smaller batch sizes lead to clearer memory of key information in dialogues), with rank and alpha set to 128 and 256, respectively. The learning rate was set to 1.0e-4, and training lasted for 3 epochs (we found this sufficient for the model to remember some dialogues, even if not achieving peak performance, ensuring fairness across different models). Detailed data formatting can be found in Appendix C.

4.2 Dataset, Models, and Metric

We utilize LOCCO as the long-term dialogue dataset and employ corresponding dialogue Q&A data to assess the model's memory. The configuration of the training data varies as we explore different research questions. Detailed data partitioning and the prompt templates used to test the model's memory with questions can be found in Appendix D.

We selected ChatGLM3-6B (GLM et al., 2024), internlm2_5-7b-chat (Cai et al., 2024), Meta-Llama-3-8B-Instruct (AI@Meta, 2024), openchat-3.5-0106 (Wang et al., 2023a), and Qwen1.5-Chat (0.5B-14B) (Bai et al., 2023) [2] as subjects of study. These models have been fine-tuned with instructions and perform well on dialogue tasks. Evaluating the response quality of generative models presents many challenges, especially when possible correct responses are diverse.

Automatic metrics like BLEU (Papineni et al., 2002) have weak correlations with human annotations, leading to significant discrepancies between different models and datasets. Some researchers use human evaluation to judge response quality, but this method is costly, time-consuming, and difficult to scale. Therefore, we trained a Consistency Model to replace human evaluation in assessing whether responses are consistent with historical

[2]Considering that the size of language model parameters might affect memory, we chose models with varying parameter sizes from the Qwen1.5-Chat series for training and testing. The Qwen1.5-Chat series offers a richer variety of models with different parameter sizes, providing a significant advantage over other series.

dialogues. More detailed training information is available in Appendix E.

We employed manual verification to validate the evaluation results of the consistency model, with the final results presented in Table 3. Detailed evaluation procedures are described in Appendix F.

Model Evaluation Results	Model Evaluation Accuracy
Consistent	94%
Inconsistent	97%

Table 3: Evaluation Accuracy of the Consistency Model.

We use response accuracy to evaluate model memory: Assume model M's response to question Q_{jx} (where Q_{jx} is a question in the set Q_j) is R_{jx}. We use A_{jx} to denote response accuracy:

$$A_{jx} = g(D_{jx}, Q_{jx}, R_{jx}) \tag{1}$$

where g represents the evaluation function. In this study, we use a consistency model as the evaluation function.If R_{jx} is consistent with the information in D_{jx}, then $A_{jx} = 1$ (meaning the model "remembers" this information). Otherwise, $A_{jx} = 0$, indicating the model "forgot" this information.

The response accuracy M_j for Q_j is:

$$M_j = \frac{1}{k} \sum_{x=1}^{k} A_{jx} \tag{2}$$

where k represents the number of questions. We use M_j to measure model M's memory of D_j. A higher M_j indicates that the model can better utilize the information in D_j to answer user questions; in other words, the higher the M_j, the stronger the model's memory of D_j.

4.3 Main Results

Long-Term Memory Performance We train the models sequentially to simulate the gradual increase in user dialogue over time, covering six time periods. After each phase, we tested the model's memory of D_1 (the initial dialogue) using Q_1. As depicted in Figure 3, all models demonstrated the highest memory retention at the outset of training. However, as training advanced, their ability to remember Q_1 generally diminished. This indicates that introducing new data makes models prone to forgetting earlier dialogue information. Within the same series, models with larger parameters (such as Qwen1.5-14B-Chat) were better at retaining early information, demonstrating stronger memory retention capabilities.

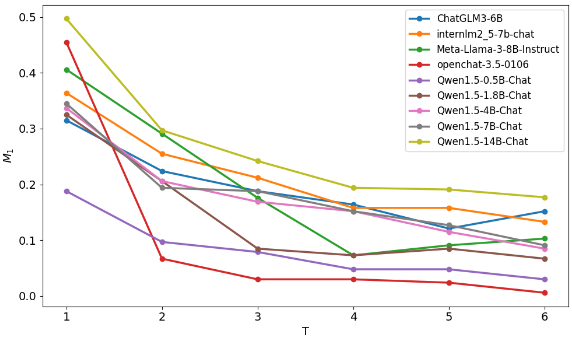

Figure 3: Memory of D_1 by LLMs at different time stages.

To more clearly observe the forgetting rate, we calculated the percentage decrease in M_1 relative to its initial value at each time point, as shown in Figure 4. Even models with similar parameter sizes (6B-8B) can exhibit significant differences in memory retention. For instance, openchat-3.5-0106 had strong memory retention at T_1 (M_1=0.455) but forgot 85.27% of the information by T_2. In contrast, ChatGLM3-6B retained 48.25% of its memory after six periods. These differences may relate to model architecture, training data, and methods.

Impact of New Data on Memory Considering that LLMs need to remember dialogues across all time periods in long-term memory tasks, we examined their ability to recall subsequent dialogue information. After training each period, we tested using corresponding dialogue Q&A. Figure 5 shows that models' memory of new dialogues gradually declines. Openchat-3.5-0106 exhibited the largest drop, with M_1 of 0.455 at T_1 falling to M6 of 0.05 at T_6, below Qwen1.5-0.5B-Chat's 0.07. ChatGLM3-6B declined more slowly, from M_1=0.31 at T_1 to M6=0.27 at T_6, a decrease of only 12.9%. While larger parameter sizes improve memory capacity, they do not mitigate the decline. Maintaining stable memory of new dialogue information is crucial for long-term tasks and remains a future challenge.

Memory Preferences Inspired by Robertson (2012), human memory for different types of information varies. We used LLMs to classify information in dialogue Q&A, with details in Appendix G. In Figure 6, We found that models exhibit varying memory strength for different categories of information, such as names, locations, and events. For instance, Llama-3-8B-Instruct had an M_1 of 0.484 for location information at T_1, 110.4% higher than for names, but location memory declined faster,

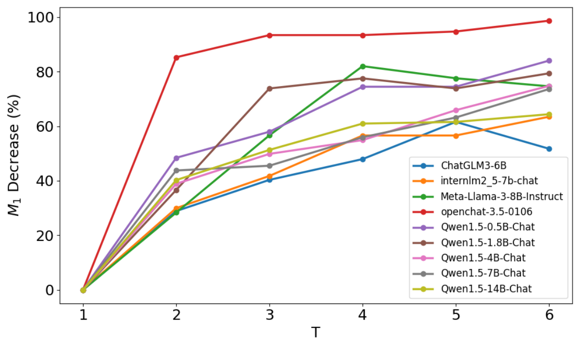

Figure 4: Percentage decrease in M_1 relative to T_1 for LLMs at different time stages. A larger M_1 Decrease indicates faster forgetting.

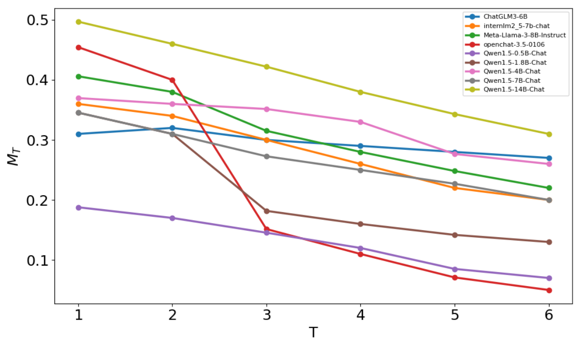

Figure 5: Memory of LLMs for new dialogues.

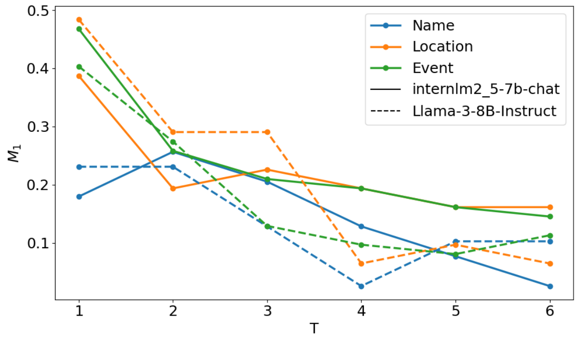

Figure 6: Memory of LLMs for different categories of information.

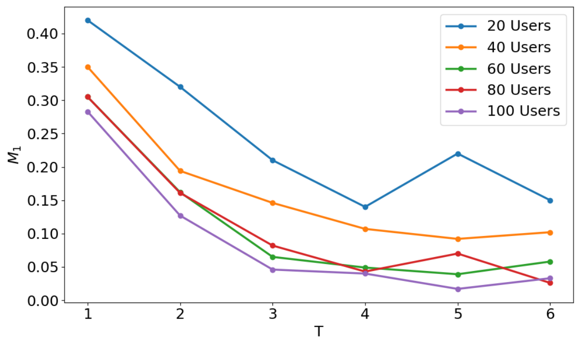

Figure 7: Impact of different dialogue densities on the long-term memory of LLMs. The model used is Qwen1.5-7B-Chat.

eventually falling below name memory. Different models also have distinct memory preferences; Llama-3-8B-Instruct remembers location information more accurately, while internlm2_5-7b-chat excels at event memory with an M_1 of 0.468. Balancing memory capabilities for different types of information can enhance long-term dialogue system performance.

Impact of Dialogue Density on Memory When LLMs need to remember a large amount of dialogue data within the same time period, their memory capabilities may also be affected. To verify this hypothesis, we selected user data of different quantities and divided the data into six time periods based on dialogue timestamps, training the models sequentially to observe the impact of dialogue density on memory performance. As shown in Figure 7, it is more challenging for the model to remember a large amount of dialogue information at once and maintain memory persistence. When the model remembers dialogues with 20 users at once, the M_1 at T_1 is 0.420, which is 48.4% higher than for 100 users (M_1 is 0.283). At T_6, the M_1 for 20-user dialogues (0.15) is 354.5% higher than the M_1 for 100-user dialogues (0.033).

Do LLMs exhibit a forgetting baseline? Tirumala et al. (2022) found that models exhibit a forgetting baseline, meaning the forgetting curve has a lower bound (the model retains a certain memory of the first batch of training data and does not completely forget). Moreover, this baseline increases with the model size, indicating that scaling up the model can mitigate forgetting. Inspired by this, we divided LOCCO into 20 time periods to observe the memory retention of LLMs over longer intervals. The experimental results are shown in Figure 8.

Notably, our experimental results differ from the observations in Tirumala et al. (2022), for long-term dialogue memory, LLMs tend to almost completely forget the initial dialogue content after a sufficiently long interval, with no memory baseline. Increasing model size does not effectively alleviate long-term forgetting.

Specifically, Tirumala et al. (2022) measures memory by evaluating the model's prediction accuracy for contexts within the training data (such as missing text segments or missing words). If a model can accurately predict the missing words, it is considered to have memorized the context. How-

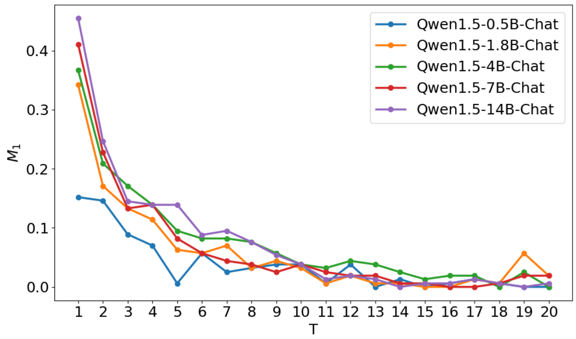

Figure 8: Forgetting of LLMs over longer time spans.

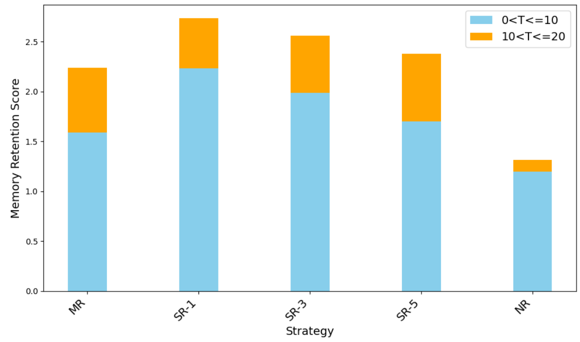

Figure 9: The impact of different repetition strategies on memory across various time ranges. MR represents Massed Repetition, SR-N represents repetition every N time periods, and NR represents no repetition. The model used is Qwen1.5-7B-Chat. We sum M_1 for the time ranges $0 < T \leq 10$ and $10 < T \leq 20$.

ever, for LLMs with reasoning capabilities, even if they do not remember the missing words, they can still infer based on existing knowledge and language structures. This leads to the model being able to guess the correct words to some extent even after forgetting all information, thereby establishing a forgetting baseline. In contrast, we assess memory by calculating the accuracy of the model's responses, thereby avoiding the aforementioned issue. Therefore, we contend that LLMs do not possess a forgetting baseline.

Replay Strategies for Permanent Memory Continual learning enables models to learn from an ongoing data stream over time. Inspired by replay strategies in continual learning (Robins, 1995; Rolnick et al., 2019; De Lange et al., 2021) as well as by the replay phenomena observed in humans (Smolen et al., 2016) and in neural network models (Amiri et al., 2017), we explore whether simple continual learning strategies remain effective for LLMs. Accordingly, we have designed the following replay strategies: i)Massed Repetition: After training on D_1, conduct three additional training sessions; ii)Spaced Repetition: Repeat D_1 within the first 10 time periods, with intervals of 1, 3, and 5 periods. Repetition is only within the first 10 periods to observe its impact on memory during and after the repetition period. We use Memory Retention Score to measure the impact of repetition on memory: summing M_1 over a specific time range represents the total memory capacity within that range. A higher score indicates stronger memory retention, as shown in Figure 9.

We find that repetition within the first 10 periods enhances memory across the entire time range, particularly in the $10 < T \leq 20$ range, showing a clear advantage over NR. Additionally, models using the SR-3 strategy outperform MR in all time ranges. Despite both undergoing three repetitions, spaced repetition is more effective than massed repetition. Moreover, we found that higher replay frequencies strengthen the model's memory within the $0 < T \leq 10$ time interval but weaken memory retention in the $10 < T \leq 20$ time interval. For LLMs, due to their vast parameter counts and complexity, continual learning differs from its application in smaller models (including smaller pretrained language models); excessive repetition is not an effective memory strategy.

5 Conclusion

To explore the long-term memory of LLMs, we developed an automated pipeline, LoCoGen, for constructing long-term dialogue data and created the LOCCO dataset, which includes long-term dialogue data between 100 users and a chatbot, along with QA pairs to evaluate model memory. We additionally constructed the LOCCO-L dataset to provide extended test cases for future research on long-term memory mechanisms. Experiments show that LLMs can remember historical interaction information with users to some extent, but this memory gradually weakens over time, especially when dealing with very long time spans. We also revealed that models have preferences when remembering different categories of information, providing a new direction for future research on how to balance and optimize memory capabilities for different types of information. Additionally, we found that repetition strategies can effectively improve the persistence of model memory. Our research not only provides new methods and datasets for evaluating the long-term memory capabilities of LLMs but also offers important references and insights for future improvements in the persistence and accuracy of model memory. Future work can further explore improvements in model architecture and training methods to better support long-term memory retention and application.

Limitations

Although the LOCCO dataset includes long-term dialogues from 100 users, these dialogues are generated by LLMs and may lack the diversity and complexity of real user interactions. Future research could incorporate more real-world data to validate the generalizability of the results. Additionally, we used closed-source models for data generation, meaning we accessed the most powerful commercial LLMs through paid APIs.

The scope of our memory testing also has certain constraints. Our current evaluation emphasizes explicit factual memory (e.g., names, places, specific events) while paying less attention to other important aspects like emotional tone retention, implicit relationship memory between characters, or the models' ability to recognize long-term behavioral patterns. These more subtle forms of memory could reveal additional insights about LLMs' memory mechanisms.

Moreover, our pipeline for generating long-term dialogues based on LLMs was developed only for English. However, our pipeline can be adapted for any other language using proficient LLMs and appropriate translations of our prompts.

References

Jaewoo Ahn, Yeda Song, Sangdoo Yun, and Gunhee Kim. 2023. Mpchat: Towards multimodal persona-grounded conversation. *arXiv preprint arXiv:2305.17388*.

AI@Meta. 2024. Llama 3 model card.

Hadi Amiri, Timothy Miller, and Guergana Savova. 2017. Repeat before forgetting: Spaced repetition for efficient and effective training of neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2401–2410.

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. *arXiv preprint arXiv:2210.08750*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Amanda Bertsch, Uri Alon, Graham Neubig, and Matthew Gormley. 2024. Unlimiformer: Long-range transformers with unlimited length input. *Advances in Neural Information Processing Systems*, 36.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. 2024. Internlm2 technical report. *Preprint*, arXiv:2403.17297.

Nuo Chen, Hongguang Li, Juhua Huang, Baoyuan Wang, and Jia Li. 2024. Compress to impress: Unleashing the potential of compressive memory in real-world long-term conversations. *Preprint*, arXiv:2402.11975.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385.

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 synthesis in question answering. *arXiv preprint arXiv:2402.16288*.

Jiazhan Feng, Qingfeng Sun, Can Xu, Pu Zhao, Yaming Yang, Chongyang Tao, Dongyan Zhao, and Qingwei Lin. 2022. Mmdialog: A large-scale multi-turn dialogue dataset towards multi-modal open-domain conversation. *arXiv preprint arXiv:2211.05719*.

Mingqi Gao, Jie Ruan, Renliang Sun, Xunjian Yin, Shiping Yang, and Xiaojun Wan. 2023. Human-like summarization evaluation with chatgpt. *ArXiv*, abs/2304.02554.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

Xu Han, Yi Dai, Tianyu Gao, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. Continual relation learning via episodic memory activation and reconsolidation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6429–6440.

Ziheng Huang, Sebastian Gutierrez, Hemanth Kamana, and Stephen MacNeil. 2023. Memory sandbox: Transparent and interactive memory management for conversational agents. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–3.

Jihyoung Jang, Minseong Boo, and Hyounghun Kim. 2023. Conversation chronicles: Towards diverse temporal and relational dynamics in multi-session conversations. *arXiv preprint arXiv:2310.13420*.

Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Stanley Jungkyu Choi, and Minjoon Seo. 2021. Towards continual knowledge learning of language models. *arXiv preprint arXiv:2110.03215*.

Zixuan Ke, Bing Liu, Nianzu Ma, Hu Xu, and Lei Shu. 2021. Achieving forgetting prevention and knowledge transfer in continual learning. *Advances in Neural Information Processing Systems*, 34:22443–22456.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, and Yejin Choi. 2023. SODA: Million-scale dialogue distillation with social commonsense contextualization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12930–12949, Singapore. Association for Computational Linguistics.

Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. 2023. How long can context length of open-source llms truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

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, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2402.17753*.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Edwin M. Robertson. 2012. New insights in human memory interference and consolidation. *Current Biology*, 22:R66–R71.

Anthony Robins. 1995. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. *Advances in neural information processing systems*, 32.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-llm: A trainable agent for role-playing. *arXiv preprint arXiv:2310.10158*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Paul Smolen, Yili Zhang, and John H Byrne. 2016. The right time to learn: mechanisms and optimization of spaced learning. *Nature Reviews Neuroscience*, 17(2):77–88.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 38274–38290. Curran Associates, Inc.

Quan Tu, Chuanqi Chen, Jinpeng Li, Yanran Li, Shuo Shang, Dongyan Zhao, Ran Wang, and Rui Yan. 2023. Characterchat: Learning towards conversational ai with personalized social support. *arXiv preprint arXiv:2308.10278*.

Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023a. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. 2023b. Huatuo: Tuning llama model with chinese medical knowledge. *arXiv preprint arXiv:2304.06975*.

Hong Wang, Wenhan Xiong, Mo Yu, Xiaoxiao Guo, Shiyu Chang, and William Yang Wang. 2019. Sentence embedding alignment for lifelong relation extraction. *arXiv preprint arXiv:1903.02588*.

Lei Wang, Jingsen Zhang, Hao Yang, Zhiyuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, Jun Xu, Zhicheng Dou, Jun Wang, and Ji-Rong Wen. 2023c. User behavior simulation with large language model based agents. *arXiv preprint*.

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, Dublin, Ireland. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2203.05797*.

Kevin Yang, Yuandong Tian, Nanyun Peng, and Dan Klein. 2022. Re3: Generating longer stories with recursive reprompting and revision. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4393–4479, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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, Singapore. Association for Computational Linguistics.

Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024. A survey on the memory mechanism of large language model based agents. *arXiv preprint*.

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 LoCoGen

A.1 Prompts

We used GPT-4o in LoCoGen to construct data, as it is one of the most powerful models currently available. For each step in LoCoGen, we initially conducted small-batch generations and manually checked the data quality, adjusting prompts to enhance the quality of the generated data. Figure 10-15 provide the prompts used in different steps.

A.2 Quality

To ensure consistent quality in LOCCO and LOCCO-L, we filtered out the following cases: (1) Dialogue data with missing or incomplete records were removed. (2) Dialogues containing excessive noise (such as spelling errors, grammatical mistakes, non-linguistic characters, etc.) were filtered out to enhance data quality and model training effectiveness. We used GPT-4o to inspect the dialogues, with specific prompts shown in Figure 16.

A.3 Human Evaluation Criteria

We require crowdworkers to evaluate the dialogue based on the following three aspects:

- Coherence: The chatbot understands the context and provides coherent responses.
- Consistency: The chatbot maintains consistency throughout the conversation.
- Participation: I enjoy interacting with this chatbot for extended periods.

Please create fictional character situations at three different time points (1 year ago, 3 years ago, 5 years ago) based on the character information provided below.

Use brief sentences to describe each time point's character situation.

Each time point must contain unique information and should reflect the alternating development of new and old things (e.g., new hobbies, further development of old interests, formation of new relationships, personality changes, etc.).

The information should be appropriate for the character's age at that time. Please describe information ("hobby", "personality", "family_relationship", "social_relationship", "study_or_work_status") in a concise paragraph:

{Character information}

Figure 10: Prompts for extending character descriptions.

Below are two character profiles from different points in time.
Please insert {N} additional profiles at different points in time between the given profiles, showcasing the progression and alternation of new and old elements (such as developing new hobbies, furthering existing interests, forming new relationships, personality changes, etc.). The profiles must fit the character's age at that time, demonstrating their development and changes to make the transitions more natural and complete. Only reply with {N} character profiles.

{Time 1 information; Time 2 information}

Figure 11: Prompts for obtaining more detailed character descriptions.

Please generate {n} coherent diary entries for the character based on the following information, with each entry occurring between the specified two time points. Each diary entry should include a date and content, and refer to the context provided to ensure coherence and consistency.
{
[Part 1: Background Information]
{Structured Data List}
[Part 2: Descriptions of specified two time points]
time1 describe:{time1 describe}
time2 describe:{time2 describe}
[Part 3: Summaries of previous diary entries]
{diaries summary}
[Part 4: Recent Diary Conten]
{last stage diaries}
}

When generating new diary entries, please follow these requirements:
{
1. Each diary entry's time point should be evenly distributed between [time1 describe] and [time2 describe].
2. The diary content should reflect the character's changes and development from time point 1 to time point 2.
3. The diary content must not conflict with the Background Information, Summaries of previous diary entries, and Recent Diary Content.
4. Each diary entry must describe a specific event, and any mentioned locations, people, or items must have specific names.
}

Figure 12: Prompts for inserting multiple events.

Please construct a multi-turn dialogue (3-5 rounds) record between a user and a chatbot based on the following the user's diary entry, with the conversation occurring at the same time as described in the diary:
{the event}
Requirements:
1. The Chatbot's responses should be conversational, logically clear, and varied.
2. The format must refer to:{formatted_data}
3. The chat must be coherent, brief and natural.

Figure 13: Prompts for generating dialogues between the user and the chatbot.

Author's past situation:
{past_elements}
Author's recent diary:
{
{events content}
}
Please update the [author's past situation] based on the [author's recent diary], ensuring the content is updated with specific descriptions for each item. For content that has changed(educational background, emotional status), keep only the most recent one.
Please output in JSON format, including [social circle list, family relationship list, study or work progress, educational background, emotional status].

Figure 14: Prompts for automatically updating the structured data list.

Please read the following diary contents and summarize all the key information from the diaries. Remove any invalid or redundant expressions, retaining only the core content of each diary. The diary contents are as follows:
{
{Events content}
}
Please output a paragraph summarizing what is discussed in all the diaries. Must be less than 500 words.

Figure 15: Prompts for summarizing event content.

Check whether the conversation data meets the following conditions. If yes, output Yes; otherwise, output No:
1. Incomplete conversations: Any missing or incomplete conversation records should be filtered out.
2. Noisy conversations: Any conversations that contain obvious noise, such as typos, grammatical errors, or non-verbal characters, should be filtered out to improve data quality and model training efficiency.
{conversation data}

Figure 16: Prompt for Dialogue Filtering.

Context:
You are an evaluator tasked with assessing the quality of a conversation between a user and a chatbot. You need to rate the conversation based on three metrics: Participation, Coherence, and Rationality.

Instructions:
Participation: Rate how actively and meaningfully both parties (user and chatbot) engage in the conversation. Consider the relevance and contribution of each turn in the dialogue.
Coherence: Evaluate the logical flow and consistency of the conversation. The dialogue should make sense as a whole, with each response appropriately following the preceding interaction.
Rationality: Assess the reasonableness and sensibility of the chatbot's responses. The responses should be logical, well-founded, and appropriate given the context of the conversation.

For each metric, provide a score on a scale from 1 to 5, where 1 is very poor and 5 is excellent.

Example Conversation:{The Conversation}

Evaluation Format:
```
{
  "Participation": [Your Score],
  "Coherence": [Your Score],
  "Rationality": [Your Score]
}
```

Figure 17: The prompt used for evaluating conversations.

A.4 Model Evaluation Criteria

We evaluated the dialogue data in terms of engagement, coherence, and plausibility. We found that data constructed by large models were of high quality. Figure 17 shows the prompts used for evaluation, and Table 4 presents the evaluation results.

B Dialogue QA Data

B.1 Generating dialogue QA pairs

Specifically, we instructed the large language model to first select a key piece of information from the dialogue and then construct a dialogue QA pair between the user and the chatbot based on this information. Key information includes names, locations, event names, etc., which are considered crucial points in the dialogue worth remembering long-term by the model. The prompts used for generating dialogue QA pairs are shown in Figure 18.

B.2 Filtering Rules

We removed QA pairs that did not meet the criteria based on the following two rules: Rule 1: The question is ambiguously phrased, leading to multiple reasonable answers. In other words, the question does not provide enough clear information, making it impossible to ensure a uniquely correct model response. Rule 2: The key information required for the answer comes from multiple different dialogue fragments. The model must rely on key information from the corresponding historical dialogue in the QA pair to answer, otherwise, it does not meet our evaluation goals.

C Training Example

To explore whether training can enable large models to remember historical dialogues, we need to construct a reasonable data format, which is different from improving the model's dialogue capability. We used supervised fine-tuning to help the large model remember dialogues with the user. Specifically, we included the character's name and dialogue timestamp as part of the instructions and used the dialogue content as labels. Specific training examples are shown in Figure 19.

D Assess Memory

D.1 Testing Example

We tested using a few-shot approach by providing the model with 3 additional correct dialogue QA examples. We found this method very effective for smaller parameter models, as their instruction-following capabilities might be insufficient to accurately comprehend test instructions. Figure 20 shows the specific prompt templates for testing memory.

D.2 Data Partition

We configure the training data differently when exploring various research questions, with the detailed data partitioning outlined below:

- Research Questions 1-3: We selected long-term dialogue data from 32 users in LOCCO and divided each user's long-term dialogues into six time periods, resulting in an average of 162 dialogues per time period. Utilizing a smaller user group helps reduce experiment duration and enhances the efficiency of model training.

- Research Question 4: We selected long-term dialogues from varying numbers of users in LOCCO and partitioned them into six time periods. The model was progressively trained to observe the impact of dialogue density, i.e., the number of dialogues per training session, on the model's memory performance.

- Research Questions 5-6: We employed long-term dialogues from all users in LOCCO and divided each user's long-term dialogues into 20 equal segments, with an average of 154 dialogues per time period.

E Training Consistency Model

When training the consistency model, we randomly selected 500 consistent responses from the QA data as positive samples and used GPT-4o to generate 500 inconsistent responses as negative samples. The dataset was split into training and validation sets in an 8:2 ratio. Training was conducted according to the instructions in Figure 21. We used Qwen1.5-4B-Chat as the pre-trained model and adopted LoRA (Low-Rank Adaptation) for training. The training process used a batch size of 4, with rank and alpha set to 128 and 256, respectively, and a learning rate of 1.0e-4, continuing for 2 epochs. A cosine annealing learning rate schedule was employed, with a 10% warm-up ratio at the beginning. Our Consistency Model achieved an accuracy of 98% on the validation set.

F Evaluating Consistency Model

We conducted manual verification of the consistency model's evaluation results. Specifically, we randomly selected 200 examples that the consistency model deemed correct and 200 examples deemed incorrect from the experimental results. Three human evaluators were then tasked with verifying the accuracy of the consistency model's assessments. The evaluators were instructed as follows: "Given a historical dialogue, a question-answer pair, and an evaluation of the answer, please determine whether the evaluation is correct. If the answer is consistent with the information mentioned in the historical dialogue, the evaluation should be consistent; otherwise, the evaluation should be inconsistent." In instances where the human evaluators' assessments differed, the majority decision was adopted.

G Classifying Information

Figure 22 shows the prompts used for classifying key information involved in the dialogue QA pairs. Table 5 displays the percentage and number of QA pairs for different categories. For categories with fewer instances, the test results may not be representative, and we merged them into the "Others" category.

Metrics	**Avg**	**Std**
Participation	4.21	0.77
Coherence	4.15	0.96
Rationality	4.42	1.02
Overall	4.26	-

Table 4: GPT-4o evaluation for the quality of LOCCO.

Category	**Percentage**	**Quantity**
Name	23.60%	704
Location	18.80%	560
Event	37.60%	1121
Others	20%	596

Table 5: Distribution of different categories.

The current time is {Currant Time}.
The following is a historical conversation between the user and the chatbot: {A history conversation}

Task: Please choose a key piece of information from the historical conversation (e.g., the name of an event, a person's name, a location, etc.), and then construct a question and answer pair between the user and the chatbot based on that key information. In the question, the user needs to provide a detailed and specific description to ensure the answer is clear and precise, guiding the chatbot to provide an accurate response based on the historical conversation.
The chatbot must use the key information mentioned in the historical conversation as part of its reply.

Please output a structured JSON object following this format:{"User": "A detailed, accurate question.","Chatbot": "Response."}

Figure 18: Prompts for Generating Dialogue QA Pairs.

#Instruction#
I am [Gabrielle Johnson], we had a chat, please recall the chat that happened on [2017-04-28].

#Output#
[
Gabrielle Johnson: Just finished at drama club. We’ re working on a new play!
Assistant: That sounds really exciting! Are you aiming for a major role?
Gabrielle Johnson: Yep, I'm hoping to get a major role again. Fingers crossed!
Assistant: I'll be rooting for you! Did you get positive vibes from your first read-through?
...
]

Figure 19: Data Format for Training Historical Dialogues.

#Example1:# I am .... Question: {User Question}

#Example2:# I am .... Question: {User Question}

#Example3:# I am .... Question: {User Question}

I am {NAME}, and the current time is {TIME}. You need to accurately recall our historical conversation , and use the information mentioned in the historical conversation to answer the question. Question: {User Question}

Figure 20: Prompts for Testing Model Memory.

Record of the conversation between the user and the chatbot: {A history conversation}
The current time is: {Currant Time}
Now, the user asks the chatbot a question to check if the chatbot remembers something mentioned in the record of the conversation:{Question}
The response of the chatbot is: {Response}
Please determine whether the response of the chatbot is accurate. If the response of the chatbot is consistent with the content in the record of the conversation, please output "Yes", otherwise output "No"."

Figure 21: Instructions for Training the Consistency Model.

Please categorize the answers to the questions. Categories need to be selected from ["people", "date and time", "location", "event", "emotions", "entity"]. You only need to output the category of the answer information.
[
{Question}
Answer:{Answer}
]
Class:

Figure 22: Prompts for classifying key information.