

Memory Sandbox: Transparent and Interactive Memory Management for Conversational Agents

Ziheng Huang

z8huang@ucsd.edu

University of California—San Diego

San Diego, CA, USA

Hemanth Kamana

tuo73589@temple.edu

Temple University

Philadelphia, PA, USA

Sebastian Gutierrez

guts@temple.edu

Temple University

Philadelphia, PA, USA

Stephen MacNeil

stephen.macneil@temple.edu

Temple University

Philadelphia, PA, USA

ABSTRACT

The recent advent of large language models (LLM) has resulted in high-performing conversational agents such as chatGPT. These agents must remember key information from an ongoing conversation to provide responses that are contextually relevant to the user. However, these agents have limited memory and can be distracted by irrelevant parts of the conversation. While many strategies exist to manage conversational memory, users currently lack affordances for viewing and controlling what the agent remembers, resulting in a poor mental model and conversational breakdowns. In this paper, we present Memory Sandbox, an interactive system and design probe that allows users to manage the conversational memory of LLM-powered agents. By treating memories as data objects that can be viewed, manipulated, recorded, summarized, and shared across conversations, Memory Sandbox provides interaction affordances for users to manage how the agent should ‘see’ the conversation.

CCS CONCEPTS

• **Computing methodologies** → **Intelligent agents**; • **Human-centered computing** → **Interactive systems and tools**;

KEYWORDS

Human-AI Interaction, Large Language Models, Chatbots

1 INTRODUCTION

Large Language Models (LLMs) are currently capable of generating human-like responses in open-domain tasks [4]. This has led to a new generation of conversational agents, such as chatGPT, which are now being widely used across domains. To ensure that agents generate responses that are contextually relevant and coherent to an ongoing conversation, these agents must maintain a working memory of the conversational history that has occurred up to that point in the conversation. The default strategy is to use as much of the conversational history as will fit within the input size limit of the LLM. Parts of the conversations that go beyond that buffer limit are forgotten, which leads to breakdowns when users assume the model remembers past context. Additionally, as the input buffer size increases, the performance of the LLM degrades as it struggles to retrieve relevant context and can be distracted by irrelevant context [11, 18]. This problem is compounded because users do not know how the LLM is leveraging the memory to generate responses.

Multiple strategies have been introduced to manage agents’ conversational memory. For example, the conversation can be automatically summarized [21] and refined [24] to reduce redundancy while maintaining key information. Some systems selectively store [12, 22] and update [1] key memories. Relevant memories can also be retrieved based on the user input [1, 15, 21]. However, these memory management strategies are hidden behind the interface, resulting in a lack of transparency. Users often do not know what strategy is being used and have limited control over it. This makes it difficult for users to repair conversational breakdowns that happen when there is a misalignment between how the agent manages the memory and how the user perceives the conversation.

We present *Memory sandbox*, shown in Figure 1, a system that allows users to see and manage the memory of conversational agents to align with user understanding of the conversation. Memory Sandbox transforms conversational memory, previously managed behind the user interface, into **interactive memory objects** within the interface. Users can manipulate the visibility and content of memory objects, spatially rearrange them, and share them across conversations. We make the following contributions: 1) The conceptualization of memory objects which makes conversational memory transparent and interactive and 2) The Memory Sandbox system that offers novel interaction affordances for users to view and manipulate the conversational memory of an intelligent agent.

2 SYSTEM OVERVIEW

Memory sandbox is a system that provides users with the ability to view and manipulate the memory model of an intelligent agent, resulting in a shared representation of their ongoing conversation. Memory Sandbox introduces the concept of a *memory object*, an interactive piece of conversational history that can be moved, edited, deleted, or combined with other memory objects through summarization. The interface is implemented in Next.js and uses the GPT-3.5 turbo model from the OpenAI API. Below we present the features of Memory Sandbox to help end users view and manage an LLM-powered agent’s memory model.

2.1 View and manipulate memory objects

Explainable AI research seeks to help people form mental models of intelligent systems [17]. Transparency of the inner workings of the system [6, 23] and interactivity to probe and manipulate the

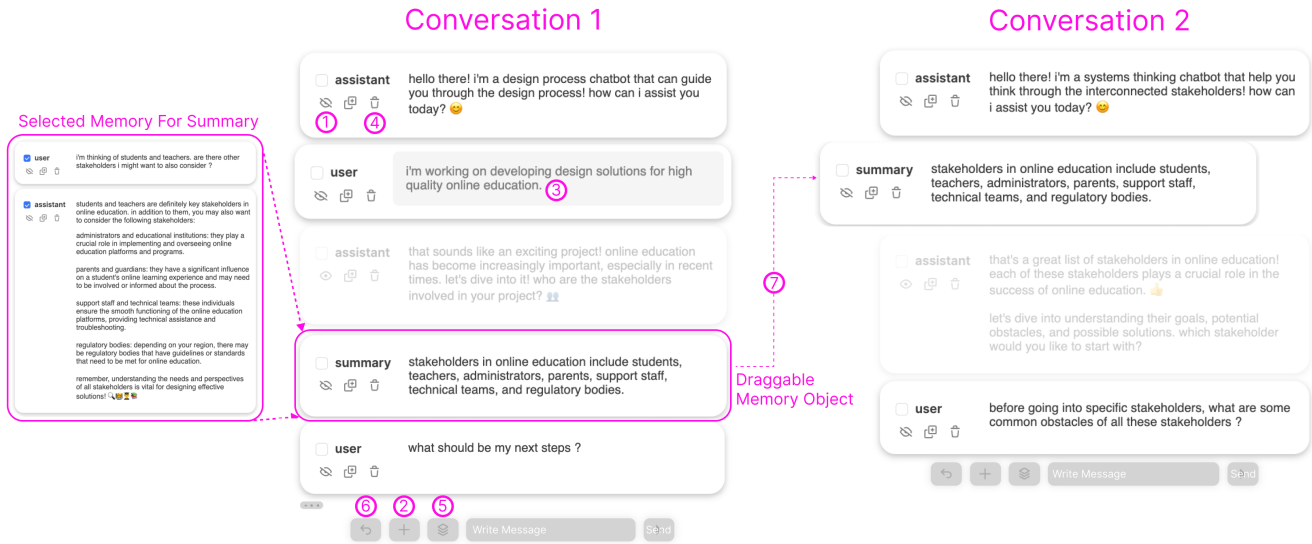


Figure 1: Memory Sandbox is a system that enables users to see and manage the memory of conversational agents. Memory Sandbox provides the following interaction affordances: 1) toggle memory visibility, 2) add memory, 3) edit memory, 4) delete memory, 5) summarize memory, 6) create a new conversation, and 7) share memory.

system [16] have been demonstrated to help people interpret and interact with intelligent systems to achieve their goals.

Memory Sandbox makes the conversational memory explicit through the use of ‘memory objects’ which can be viewed and manipulated within the interface. This was inspired by prior work that ‘objectifies’ tools [2, 3] and attributes [20] to enable flexibility, expressiveness, and direct manipulation. This results in a ‘shared representation’ [7, 8] and common ground [5]—so what users see on the front-end is what an LLM would ‘see’ on the back-end.

Additionally, users can view, edit, add, and delete memory objects to directly control how the agent ‘sees’ the conversation.

2.2 Toggle memory object visibility

As a conversation grows, LLMs must increasingly rely on their memory management strategy to infer meaning from the conversation. However, in longer conversations, it is unclear what parts of the conversation are stored in memory or are attended to by the model [11]. This results in a poor mental model for users and a lack of control over what context is maintained and used by the agent.

Memory Sandbox enables users to selectively hide or show memory objects to control what context is shared with the agent. When the user’s intent changes or the conversational context switches, the user can toggle the visibility of memory objects to hide or show parts of the conversation. As a signifier, hidden memory objects are grayed out within the interface.

2.3 Curate memory objects

Discussions develop and refine their understanding as a conversation unfolds [5]. Thus, Memory Sandbox provides controls for users to curate memory objects by editing an existing memory object to refine or update the context, deleting a memory object to remove completely irrelevant context, and adding a new memory object to

supplement extra context. Additionally, the arrangement of context is shown to have a significant effect on how well LLMs are able to leverage relevant context [11]. In Memory Sandbox, all the memory objects are draggable, allowing users to experiment and refine the ordering and placement of memory objects in a conversation.

2.4 Summarize memory objects

Reminiscent of how humans attend to key aspects in a conversation [14], abstractive summarization distills a large amount of information to provide essential elements to the agent. Yet, what is considered as ‘key aspects’ can vary for individuals, even in the same conversation [14]. Memory Sandbox enables users to select memory objects that are summarized by the LLM. The resulting memory object represents the previous conversation and can be further refined by the user. The original conversation can be viewed by clicking on the summary.

2.5 Share memory objects across conversations

Aligning with the goal of managing memory, Memory Sandbox also provides affordances for sharing memories across conversations. This offers a new way for users to engage with multiple agents outside of a single conversation thread. Unlike in conversations with people, the speaker doesn’t need to repeat themselves in each conversation to establish a shared understanding.

Users can create and start multiple conversations with separate LLM-powered agents in the same 2D canvas. Memory objects can be shared and connected between conversations by dragging the memory object from one conversation to another. When dragging, memories are copied by reference to help the user identify the context source.

3 DISCUSSION

Conversing is a collaborative activity where participants develop common ground through summarizing the discussion, repairing breakdowns, and emphasizing or de-emphasizing shared ideas [5]. Yet, existing chatbot interfaces do not provide affordances for understanding how the agent ‘sees’ the conversation. Additionally, users can not rely on a theory of mind. These aspects result in a poor mental model for users and potential misalignment in understanding where conversational breakdown can occur.

Memory Sandbox transforms previously implicitly managed conversational memory behind the interface into interactive memory objects on the interface, exposing full control over the memory model of the agent to end users. By selectively hiding, showing, and curating memory representation, we can give users more control over how the agent should “see” the conversation. In addition to curating memory in a single conversation, Memory Sandbox is also a design probe toward memory manipulation affordances for multi-agent interactions. By displaying multiple agents on the same screen and making memories interactive and draggable, Memory Sandbox allows end users to selectively control the shared or unique memory each agent contains.

Tools are beginning to emerge that focus on how users might interact with LLMs, including mapping UI affordances to an LLM [13], grounding human-AI collaboration in a shared artifact [9], providing templates to facilitate prompt generation [10], and decomposing complex prompts to facilitate debugging [19]. In this paper, we presented Memory Sandbox an interactive system that probes the design space of interaction techniques for memory management of LLMs. Our future work includes user studies to evaluate the efficacy of these techniques and potential trade-offs for implicit vs explicit memory management

REFERENCES

- [1] Sanghwan Bae, Donghyun Kwak, Soyoung Kang, Min Young Lee, Sungdong Kim, Yuiin 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* (2022).
- [2] Benjamin B Bederson, James D Hollan, Allison Druin, Jason Stewart, David Rogers, and David Proft. 1996. Local tools: An alternative to tool palettes. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*. 169–170.
- [3] Eric A Bier, Maureen C Stone, Ken Pier, William Buxton, and Tony D DeRose. 1993. Toolglass and magic lenses: the see-through interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. 73–80.
- [4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
- [5] Herbert H Clark and Edward F Schaefer. 1989. Contributing to discourse. *Cognitive science* 13, 2 (1989), 259–294.
- [6] Malin Eiband, Hanna Schneider, Mark Bilandzic, Julian Fazekas-Con, Mareike Haug, and Heinrich Hussmann. 2018. Bringing transparency design into practice. In *23rd international conference on intelligent user interfaces*. 211–223.
- [7] Jeffrey Heer. 2019. Agency plus automation: Designing artificial intelligence into interactive systems. *Proceedings of the National Academy of Sciences* 116, 6 (2019), 1844–1850.
- [8] Eric Horvitz. 1999. Principles of Mixed-Initiative User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) (CHI ’99). Association for Computing Machinery, New York, NY, USA, 159–166. <https://doi.org/10.1145/302979.303030>
- [9] Ziheng Huang, Kexin Quan, Joel Chan, and Stephen MacNeil. 2023. CausalMapper: Challenging designers to think in systems with Causal Maps and Large Language Model. In *Proceedings of the 15th Conference on Creativity and Cognition*. 325–329.
- [10] Ellen Jiang, Kristen Olson, Edwin Toh, Alejandra Molina, Aaron Donsbach, Michael Terry, and Carrie J Cai. 2022. Promptmaker: Prompt-based prototyping with large language models. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*. 1–8.
- [11] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranajpe, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the Middle: How Language Models Use Long Contexts. *arXiv:2307.03172 [cs.CL]*
- [12] Zhengyi Ma, Zhicheng Dou, Yutao Zhu, Hanxun Zhong, and Ji-Rong Wen. 2021. One chatbot per person: Creating personalized chatbots based on implicit user profiles. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 555–564.
- [13] Stephen MacNeil, Andrew Tran, Joanne Kim, Ziheng Huang, Seth Bernstein, and Dan Mogil. 2023. Prompt Middleware: Mapping Prompts for Large Language Models to UI Affordances. *arXiv preprint arXiv:2307.01142* (2023).
- [14] John C McCarthy, Victoria C Miles, and Andrew F Monk. 1991. An experimental study of common ground in text-based communication. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 209–215.
- [15] Joon Sung Park, Joseph C O’Brien, Carrie J Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442* (2023).
- [16] Andrew Ross, Nina Chen, Elisa Zhao Hang, Elena L Glassman, and Finale Doshi-Velez. 2021. Evaluating the interpretability of generative models by interactive reconstruction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.
- [17] Heleen Rutjes, Martijn Willemsen, and Wijnand IJsselstein. 2019. Considerations on explainable AI and users’ mental models. In *CHI 2019 Workshop: Where is the Human? Bridging the Gap Between AI and HCI*. Association for Computing Machinery, Inc.
- [18] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*. PMLR, 31210–31227.
- [19] Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. In *Proceedings of the 2022 CHI conference on human factors in computing systems*.
- [20] Haijun Xia, Bruno Araujo, Tovi Grossman, and Daniel Wigdor. 2016. Object-oriented drawing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 4610–4621.
- [21] Jing Xu, Arthur Szlam, and Jason Weston. 2021. Beyond goldfish memory: Long-term open-domain conversation. *arXiv preprint arXiv:2107.07567* (2021).
- [22] Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. 2022. Long Time No See! Open-Domain Conversation with Long-Term Persona Memory. In *Findings of the Association for Computational Linguistics: ACL 2022*. Association for Computational Linguistics, Dublin, Ireland, 2639–2650. <https://doi.org/10.18653/v1/2022.findings-acl.207>
- [23] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. 2020. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 295–305.
- [24] Hanxun Zhong, Zhicheng Dou, Yutao Zhu, Hongjin Qian, and Ji-Rong Wen. 2022. Less is more: Learning to refine dialogue history for personalized dialogue generation. *arXiv preprint arXiv:2204.08128* (2022).