



OPEN ACCESS

EDITED BY

Kyle Perkins,
Florida International University, United States

REVIEWED BY

Alain Lioret,
Université Paris 8, France
Susan Neimand,
Miami Dade College, United States

*CORRESPONDENCE

Qingyun He
hqy@ahstu.edu.cn

RECEIVED 11 March 2025

ACCEPTED 11 April 2025

PUBLISHED 07 May 2025

CITATION

Hong C and He Q (2025) Enhancing memory retrieval in generative agents through LLM-trained cross attention networks. *Front. Psychol.* 16:1591618.
doi: 10.3389/fpsyg.2025.1591618

COPYRIGHT

© 2025 Hong and He. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](#). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Enhancing memory retrieval in generative agents through LLM-trained cross attention networks

Chuanyang Hong¹ and Qingyun He^{2*}

¹School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu, China, ²School of Finance and Economics, Anhui Science and Technology University, Bengbu, China

Introduction: The surge in the capabilities of large language models (LLMs) has propelled the development of Artificial General Intelligence (AGI), highlighting generative agents as pivotal components for emulating complex AI behaviors. Given the high costs associated with individually training LLMs for each AI agent, there is a critical need for advanced memory retrieval mechanisms to maintain the unique characteristics and memories of individual AI agents.

Methods: In this research, we developed a text-based simulation of a generative agent world, constructing a community with multiple agents and locations in which certain levels of interaction were enabled. Within this framework, we introduced a novel memory retrieval system using an Auxiliary Cross Attention Network (ACAN). This system calculates and ranks attention weights between an agent's current state and stored memories, selecting the most relevant memories for any given situation. In a novel approach, we incorporated LLM assistance, comparing memories retrieved by our model with those extracted using a base method during training, and constructing a novel loss function based on these comparisons to optimize the training process effectively. To our knowledge, this is the first study to utilize LLMs to train a dedicated agent memory retrieval network.

Results: Our empirical evaluations demonstrate that this approach substantially enhances the quality of memory retrieval, thereby increasing the adaptability and behavioral consistency of agents in fluctuating environments.

Discussion: Our findings not only introduce new perspectives and methodologies for memory retrieval in generative agents but also extend the utility of LLMs in memory management across varied AI agent applications.

KEYWORDS

artificial intelligence (AI), large language models (LLMs), generative agents, memory retrieval, attention mechanism

1 Introduction

The release of GPT-4 by OpenAI has demonstrated the impressive capabilities of large language models (LLMs) and their potential for Artificial General Intelligence (AGI). Consequently, various Artificial Intelligence (AI) applications based on LLMs have made significant advancements across different fields. Among these, personalized AI agents that simulate human behavior have garnered increasing attention and are considered a crucial pathway toward AGI (Xi et al., 2023).

The concept of an agent refers to entities possessing desires, beliefs, intentions, and the ability to take actions (Zalta et al., 1995). Currently, the goal of LLM based generative agents is to simulate believable human behavior, creating more personalized AI. This requires AI not only to simulate human behavior at a single point in time but to ensure long-term coherence. Such AI would be better suited by architectures that manage ever-growing memories as new interactions, conflicts, and events arise and fade over time while handling cascading social dynamics that unfold between multiple agents (Park et al., 2023).

Therefore, personalized AI requires not only the general intelligence provided by LLMs but also long-term personalized memories that are private, extensible, and explainable to the user. Additionally, it requires an efficient method to retrieve these relevant memories based on the current context faced by the agent.

To achieve this goal, the ideal approach would be to train a dedicated LLM for each agent. However, considering the complexity of LLM training (Yang et al., 2024) and the practical demands of a large variety and number of agents, this approach is impractical. Therefore, the common practice is to store the agent's memories externally and provide the necessary memories to the LLM in the form of linguistic feedback during decision-making (Shinn et al., 2023).

In this approach to implementing agents, the method of memory retrieval becomes critically important. The ability to extract memories relevant to the current context faced by the agent will directly determine how well the agent's behavior can simulate real human actions. Common memory retrieval methods include temporal decay ranking, evaluation of memory importance, vector similarity matching, and combinations of these techniques (Park et al., 2023). However, these existing methods still have significant limitations in matching the complex correlations between the agent's current context and the memories stored in the memory bank.

Faced with this challenge, we developed a text-based generative agent simulation environment featuring multiple characters and locations, as depicted in Figure 1. This simulation framework enabled the modeling of agents with diverse characteristics, including varying ages, genders, identities, professions, and personalities, all portrayed by LLMs. These agents operated within a virtual village, residing in their respective homes and interacting in public spaces. Through extended simulations and systematic observation of the agents' behaviors and feedback, we sought to evaluate the impact of different memory retrieval methods on the agents' ability to simulate human behavior effectively.

Building on this foundation, we propose an innovative memory retrieval method designed for generative agents that simulate human-like interactions. This method uses an Auxiliary Cross Attention Network (ACAN) to optimize memory retrieval. Inspired by the self-attention mechanism described in Vaswani et al. (2017), ACAN transforms the agent's current state and observed context into a query vector. This query is compared with stored memories in the memory bank, which are represented as key-value pairs. The attention mechanism calculates scores by aligning the query with the memory keys, and the attention weights are ranked. Based on these ranked attention scores,

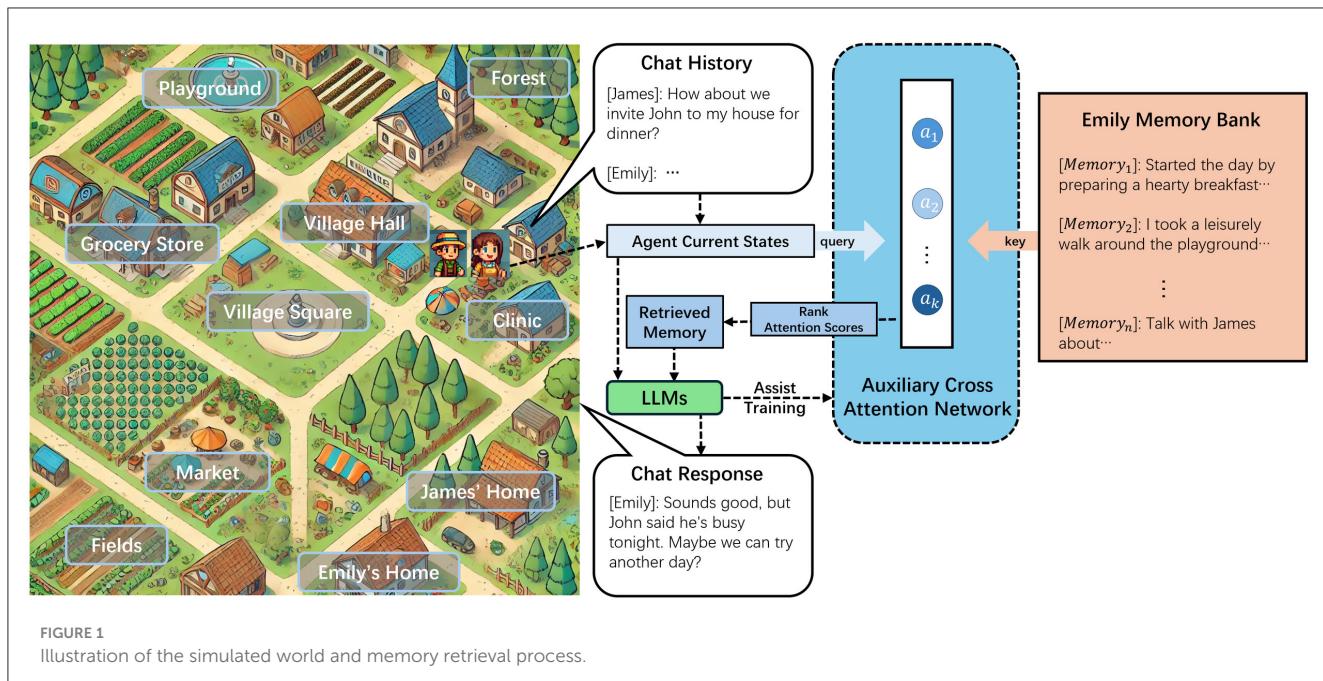
the most relevant memories are selected as the retrieved memory set.

The retrieved memories are combined with the agent's current state and input into the LLM to guide the agent's behavior. To train this network and enhance its ability to simulate the human memory retrieval process, we innovatively introduced the use of LLMs to assist in network training. By comparing the memories retrieved by this network with traditional memory retrieval methods, allowing the LLM to evaluate and score the quality of the retrieved memories based on the agent's current state. These scores are then incorporated into the custom loss function to guide the training of the ACAN. This method ensures that the network is updated in a way that better reflects human-like memory retrieval patterns. To the best of our knowledge, this is the first approach that integrates LLMs into the training process of a memory retrieval network for agents.

Compared to existing static memory retrieval algorithms, the ACAN approach introduces dynamic improvements by incorporating the agent's historical memory formation process with a cross-attention mechanism that is optimized through LLM feedback. Our experiments demonstrate that ACAN substantially outperforms traditional methods in memory retrieval, resulting in enhanced adaptability of agents and more effective interactions with their environment and other agents. We evaluated the quality of the retrieved memories using LLMs in a comprehensive test simulation set and conducted a quantitative analysis of agent behavior consistency across various memory retrieval modes. This novel memory retrieval method allows agents to better simulate human-like responses based on their current state, thereby significantly improving their ability to engage in complex interpersonal interactions.

In summary, our paper makes the following contributions:

- We constructed a novel, text-based generative agent simulation environment, featuring multiple characters and locations, which simulates real-life interactions at a low computational cost, demonstrating a novel application of LLMs in simulating human-like agent behavior.
- We introduced an innovative Auxiliary Cross Attention Network for memory retrieval in AI agents simulating human behavior. By calculating attention weights between the agent's current state and all memories in the memory bank, ACAN ranks and retrieves the most relevant memories, leading to Enhanced Memory Retrieval compared to base methods.
- We introduced a novel methodology for training neural networks with LLM assistance, where LLMs evaluate memory retrieval outputs and help in shaping the loss function. This innovative use of LLMs in training AI agents offers a fresh perspective on their application in the AI agent field.
- We provide a detailed comparison between our method and commonly used memory retrieval techniques, demonstrating our approach's superior ability to dynamically adapt to the agent's evolving memory and environment. This novel memory retrieval method enables agents to more accurately simulate human-like responses based on their current state, significantly enhancing their capacity to engage in complex interpersonal interactions.



2 Related work

2.1 Large language models and generative agent

Generative artificial intelligence refers to AI systems that generate text, images, videos, or other data types based on prompts. These systems are exemplified by LLMs such as ChatGPT and GPT-4, which have achieved tremendous success across various tasks in the field of Natural Language Processing (NLP) (OpenAI, 2022). The primary feature of LLMs is their use of large-scale datasets to train large-scale models, such as GPT-3 (Brown et al., 2020), a precursor to ChatGPT, which was trained using massive data and the Transformer architecture (Vaswani et al., 2017).

The success of OpenAI's ChatGPT has sparked considerable interest among researchers as a potential spark for Artificial General Intelligence (AGI) (Xi et al., 2023). Numerous studies have validated the exceptional performance of LLMs when appropriately prompted in downstream tasks, showcasing their versatility and intelligence (Bang et al., 2023; Wei et al., 2022). These models have been effectively employed in a variety of applications, such as translation (Jiao et al., 2023), text generation across diverse genres (Cao et al., 2023), and narrative content adaptation (Musacchio et al., 2024).

A particularly notable application is the development of AI agents capable of mimicking human behaviors using ChatGPT, which illustrates the models' capability to generate believably human-like interactions (Xi et al., 2023).

In the broadest sense, an "agent" is defined as an entity capable of action (Zalta et al., 1995). Within the field of artificial intelligence, there has been a longstanding commitment to using agents as believable proxies for human behavior, a goal that holds significant importance across AI and its applications (Bates et al., 1994; Laird and VanLent, 2001; Yannakakis, 2012). Typically,

AI-based agents are designed to perceive their environments through sensors, make decisions, and perform actions using effectors (Wooldridge and Jennings, 1995; Russell and Norvig, 2016). These agents combine sensory data with pre-programmed behaviors to interact with their surroundings effectively, but creating agents that truly mimic the nuanced behaviors of humans remains a complex endeavor.

Despite this, the development of AI agents that can accurately and credibly simulate complex human behaviors has proven to be challenging (Schweitzer et al., 2020; Abdalla and Mishra, 2021). Currently, the widespread success of LLMs, exemplified by GPT-4, across various AI domains (Achiam et al., 2023), has enabled these models to leverage extensive training data on human behavior (Brown et al., 2020), providing agents with enhanced creativity and adaptability. This capability enables agents to process information more effectively and respond in ways that closely mimic human reactions. Consequently, an increasing number of researchers are exploring the use of LLMs to develop Generative Agents with robust content generation capabilities (Park et al., 2023; Liu et al., 2023; Wang et al., 2024). These agents are finding applications in diverse fields, demonstrating the versatility and potential of LLM-driven agent applications.

LLM-based agents primarily utilize prompt chains (Wu T. et al., 2022) to generate concise natural language descriptions and actions for characters within prototype systems, thereby creating populated prototypes for social computing systems (Park J. S. et al., 2022). Additionally, LLMs are employed to craft interactive experiences in user-engaging games, facilitating dynamic actions (Freiknecht and Effelsberg, 2020) and text-based adventure games (Callison-Burch et al., 2022). Further extending their application, LLM-driven Generative Agents are used to construct virtual communities. Within these simulated environments, researchers have observed social phenomena emerging from the cooperation among multiple agents (Park et al., 2023). For instance, in a virtual community,

an agent planning a Valentine's Day party autonomously spreads invitations throughout the community and coordinates the timing of the event over the following two days.

To enable Generative Agents, assisted by LLMs, to perform such complex functions, researchers have explored methods beyond first-order prompting. They have enhanced language models with static knowledge bases and information retrieval schemes (Khattab et al., 2022), and extended these concepts to develop agent architectures that dynamically update past experiences at each step, integrating these with the agents' current contexts and plans. For instance, applications in various domains utilize such memory-enhanced agents to process layered information and improve decision-making (Yu et al., 2024). This integration can either reinforce or contradict the ongoing interactions, providing a more adaptive and responsive agent behavior (Park et al., 2023).

However, the complex behavior of agents inevitably leads to challenges similar to human decision-making, particularly the need for an appropriate memory system. This system must enable agents to retrieve the most relevant memories when needed, thereby facilitating recollection and thought processes akin to human cognition. Without such a system, agents may exhibit inconsistent behaviors over time, undermining the believability and effectiveness of their interactions.

2.2 Agent memory retrieval

In constructing memory systems for Generative Agents, agents' memories—comprised of sequences of past observations, thoughts, and actions (Nuxoll and Laird, 2007)—play a crucial role in strategy formulation and decision-making processes. Just as the human brain utilizes prior experiences for adaptive behavior (Squire, 1986; Schwabe et al., 2014), agents require specialized memory mechanisms to effectively manage sequential tasks. Research by Schuurmans (2023) demonstrated that transformer-based large language models (LLMs), when augmented with external memory, achieve computational universality. This augmentation allows agents to revisit and reapply past strategies without altering the model's weights, which is critical for reliable adaptation in complex environments.

Before the advent of LLM-based agents, extensive research had already been conducted on enhancing model performance through memory mechanisms. For instance, Memory Transformer and Recurrent Memory Transformer (Burtsev et al., 2020; Bulatov et al., 2022) introduced memory tokens and recurrent mechanisms to improve transformers' understanding of long-sequence tasks, especially for global context processing. Memorizing Transformers (Wu Y. et al., 2022) leveraged non-differentiable memory lookup systems to retrieve past inputs during inference, enabling real-time memory retrieval. Additionally, hardware-related research has explored optimizing memory utilization to improve model efficiency (Sridharan et al., 2023). However, these memory mechanisms primarily targeted deep learning models, optimizing performance through memory augmentation or architectural adjustments within a fixed model framework.

In contrast, LLM-derived agents, functioning as independent entities, face a more complex and dynamic memory landscape. These agents do not rely solely on internally generated representations from training, but also draw heavily from their interaction history and external memory repositories. For example, Memory Sandbox (Huang et al., 2023) introduced a system where users can manage conversational memories of LLM-powered agents, treating them as data objects that can be viewed, manipulated, and controlled, thus enhancing interaction transparency and coherence. Similarly, AgentSims (Lin et al., 2023) provided a sandbox infrastructure for task-based evaluations of LLM agents in simulated environments, giving researchers a platform to test memory and planning mechanisms in LLMs. A recent survey (Zhang et al., 2024) further highlights the significance of memory modules in enabling LLM-based agents to achieve self-evolving capabilities and interact effectively in real-world contexts. Furthermore, the Retrieval-Augmented Planning (RAP) framework (Kagaya et al., 2024) leverages contextual memory to enhance decision-making in both text-based and multimodal environments.

Enhancing memory retrieval in generative agents not only improves LLM performance but also enhances the extraction of external memories, thereby boosting the agents' behavior and adaptability. This is particularly critical in multi-agent systems, where each agent may have distinct external memory structures, making efficient retrieval essential. The primary method for memory utilization in LLM-based agents involves using relevant memories as prompts. However, as agents accumulate more historical data through interactions, two major challenges arise. First, the length of these records may exceed the processing limits of the LLM's Transformer architecture, causing content truncation. Second, the growing volume of observations and actions complicates the retrieval of relevant memories, leading to potential misalignment between the agent's responses and the current context. Addressing these challenges requires the development of efficient memory retrieval systems capable of managing and utilizing extensive historical data in a way that maintains coherence and relevance in the agent's interactions.

To address these challenges, current improvements in agent memory management include techniques such as text truncation (Park H. H. et al., 2022), input segmentation (Mohtashami and Jaggi, 2023), and other approaches aimed at reducing complexity, such as increasing the sequence length limits of Transformer-based LLMs (Guo et al., 2021), or incorporating self-controlled memory systems to manage long-term and short-term memory efficiently (Liang et al., 2023). Furthermore, methods for integrating and summarizing memories to create condensed representations have been developed (Zhao et al., 2024; Liang et al., 2023), enhancing the efficiency of memory retrieval in dynamic and complex interaction scenarios. Retrieval models such as Alonso et al. (2024) integrate chained-of-table search, vector-database retrieval, and prompting mechanisms to handle time-sensitive and context-dependent queries. Similarly, Hou et al. (2024) propose a human-like memory architecture for LLM-based dialogue agents, leveraging cue-based recall and a mathematical model for dynamic memory consolidation, enabling temporal and context-sensitive retrieval. Additionally, data structures and embedding techniques have been explored to compress memories, facilitating faster

response times in interactions (Modarressi et al., 2023; Qian et al., 2023), while SQL-integrated systems enable efficient management of large-scale historical data through SQL commands (Hu et al., 2023; Zhou et al., 2023).

In multi-agent systems, when agents interact with their environment and other agents, the ability to retrieve the most relevant information from their memory is essential. Particularly in environments that require collaboration among multiple agents, the quality of memory retrieval significantly influences the agents' decision-making, actions, and adaptability. This crucial aspect of memory optimization is aligned with the objectives of multi-agent reinforcement learning (MARL), where enhancing agent capabilities is a primary focus (Gronauer and Diepold, 2022). For example, the introduction of memory-driven communication mechanisms via memory devices has enabled agents to share and update information about their environment during task execution, significantly improving coordination and performance in complex multi-agent systems (Pesce and Montana, 2020).

However, unlike traditional MARL approaches that primarily utilize memory for storing learned policies or state-action histories, LLM-based multi-agent systems rely on pre-trained models, and their intelligence is not updated through real-time training as in MARL. In MARL, agents continuously improve by interacting with their environment, refining their strategies via reinforcement learning. In contrast, LLM agents depend on external, evolving memory banks to access accumulated historical interactions. The focus thus shifts from real-time learning to optimizing memory retrieval, as these external memories are queried in real-time. ACAN enhances LLM agents by improving how relevant memories are retrieved, allowing for more effective decision-making and adaptability in complex environments. MemoryBank (Zhong et al., 2024) exemplifies this, using past interaction data and the forgetting curve theory to optimize memory retrieval. Similarly, advanced methods use metrics like Recency, Relevance, and Importance to dynamically rank and retrieve the most suitable memories (Park et al., 2023), underscoring the importance of adaptive memory systems in evolving agent environments.

In summary, the literature review underscores the critical role of memory in enhancing the capability and adaptability of agents within multi-agent systems. The efficacy of generative agents in practical applications is directly determined by the capability of memory retrieval systems to extract the most relevant memories from the memory bank, akin to human-like recollection based on the current context faced by the agent. However, current methods of memory retrieval still struggle to perfectly extract the most relevant memories from the memory bank as a human would, based on the agent's current scenario.

3 Methods

To validate the effectiveness of our proposed Auxiliary Cross Attention Network for agent memory retrieval, we have structured the experimental section into distinct parts. The first part details the operational architecture of our text-based generative agent community, which is powered by ChatGPT. The second part describes the structure and training methodology of the Auxiliary

Cross Attention Network. Together, these sections provide a comprehensive overview of the experimental framework.

3.1 Generative agent architecture

To construct a virtual agent community for testing memory retrieval mechanisms, we simplified the structure described in Park et al. (2023) and proposed a purely text-based community architecture without visual imagery. This setup allows for the instantiation of maps and unique agent entities, where the locations on the map and the number and characteristics of each agent, including their professions and personalities, can be freely defined. We adopted the same GPT-3.5-turbo version of ChatGPT used in Park et al. (2023) to generate agent behaviors. This approach ensures that the comparison of memory retrieval performance is not influenced by variations in the capabilities of the large language models.

In our community architecture, we designated eight agents, each assigned a representative occupation: Farmer, Grocer, Doctor, Mayor, Chef, Hunter, Blacksmith, and Carpenter. Each agent was also given a personality description relevant to their profession. This configuration enriches the complexity of the agent community while balancing the time required for simulation.

Furthermore, in terms of the map design, we allocated a specific home and workplace for each agent, along with a corresponding functional description to ensure alignment with the agent's profession. For instance, the agent with the occupation "Farmer" has "Fields" as their workplace, while the "Doctor" is associated with the "Clinic" as their workplace. Beyond the individual homes and workplaces of each agent, the map includes several communal locations such as the "Village Square" and "Playground" to enhance interactions among the agents.

As illustrated in Figure 2, to streamline the simulation process, we structured the map and temporal dimensions using a discretized, turn-based format. Temporally, we divided each day within the simulated community into 16 active hours, from 6:00 AM to 9:00 PM, with the remaining hours allocated for sleep. During each hour, agents sequentially act based on their current states and observations. They plan their actions, decide whether to interact with other agents at the same location, and upon completing their actions, they determine their next destination based on the outcomes and their current states. This setup ensures a controlled environment where the impact of agent interactions and decision-making processes can be methodically analyzed.

During the simulation process, memory preservation and retrieval are integral to every action undertaken by an agent. Each time an agent's plan or action is determined through a prompt processed by the LLM, it requires extracting relevant memories from the agent's memory bank using a memory retrieval algorithm. This retrieval is based on the current state query to decide the agent's subsequent actions. Additionally, after each action, agents store their experiences, actions, and observations back into the memory bank, enhancing the resources available for future memory retrieval. Thus, under these simulation conditions, the critical role of the memory retrieval algorithm is further emphasized,

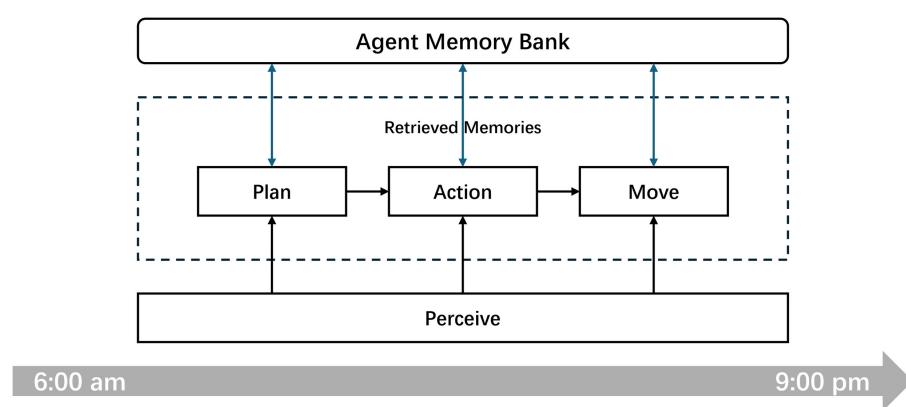


FIGURE 2

Discrete time cycle of agent activities in the simulated community. The diagram illustrates the daily schedule from 6:00 AM to 9:00 PM, delineating the key activities: planning, action, and movement, underpinned by continuous perception.

highlighting its importance in the functionality and effectiveness of the generative agents.

3.2 Auxiliary cross attention network for memory retrieval

In this section, we outline our methodological approach step by step, including the algorithmic details necessary to train the Auxiliary Cross Attention Network for Memory Retrieval. To commence training, we must first generate the requisite training data through simulation. Both the simulation and the subsequent training phases require a foundational memory retrieval system to facilitate the agent's memory recall processes effectively.

Existing memory retrieval methods for agents primarily focus on relevance and temporal validity. For instance, Hou et al. (2024) propose a human-like memory architecture with cue-based recall and dynamic memory consolidation. Retrieval models such as Alonso et al. (2024) employ vector-database mechanisms to handle time-sensitive and context-dependent queries. Similarly, Park et al. (2023) introduce a generative memory scoring framework that balances multiple retrieval criteria. To address these aspects comprehensively, we have implemented a unified memory scoring method, Weighted Memory Retrieval (WMR), as our baseline memory retrieval approach, which calculates memory retrieval scores based on the following criteria:

$$\text{WMR}(m) = w_r \cdot \text{Recency}(m) + w_i \cdot \text{Importance}(m) + w_s \cdot \text{Relevance}(m, q), \quad (1)$$

In this formulation, *Recency* represents the memory decay score, which decreases hourly by a decay factor of 0.995. The *Importance* score is generated by LLM, determining the agent's perceived significance of the memory. *Relevance* measures the cosine similarity between the embedding vectors of each memory in the memory bank and the current state's query embedding. This is mathematically expressed as:

$$\text{Relevance}(m, q) = \frac{m \cdot q}{|m||q|} \quad (2)$$

where m is the memory embedding vector and q is the query embedding vector.

After applying this scoring system, the memories with the highest scores are selected, and the top k memories are retrieved as the base memory set m'_r .

During the training of our Auxiliary Cross Attention Network, as detailed in Algorithm 1, we systematically employ a dataset consisting of the states q faced by each agent during decision-making and the associated memory bank M collected during the simulation. The decision-making contexts and the corresponding memory banks of each agent are converted into high-quality text embeddings using the text-embedding-ada-002 model provided by OpenAI (Neelakantan et al., 2022). This model ensures the embeddings preserve the semantic richness essential for effective training. The algorithm iteratively adjusts the network weights to optimize the retrieval of relevant memories based on the agents' current contextual needs. This optimization is facilitated by a cross attention mechanism that aligns the agent's query with the most relevant information from the memory bank.

Furthermore, drawing upon research in explainable deep learning (Serrano and Smith, 2019), we innovatively decided to determine the output of the model, specifically the retrieved memories, not by the weighted memories vector m_{weighted} , but rather through the model's cross attention mechanism. The computation of attention probabilities is defined by the following equation:

$$\mathbf{P} = \text{Softmax} \left(\frac{\mathbf{q}_{\text{query}} \cdot \mathbf{m}_{\text{key}}^T}{\alpha} \right) \quad (3)$$

This change emphasizes the importance of interpretability in memory retrieval, allowing for a clearer understanding of how and why certain memories are retrieved based on the agents' current queries.

```

Input: Training dataset consisting of query set  $q$ ,  

        associated memory bank  $M_q$ , and WMR  

        retrieved memories  $m'_r$   

Output: Retrieved memories  $m_r$  based on attention  

        mechanism

1 Initialize:
    • Generate text embeddings for queries  $q \in Q$  and  

        memories  $m \in M_q$  using the text-embedding-ada-002  

        model from OpenAI.  

    • Set scale factor  $\alpha = \sqrt{\text{embed\_size}}$ .  

for each training iteration do
    for each query  $q$  in  $Q$  do
         $\mathbf{q}_{\text{query}} = \text{QueryNetwork}(q)$ ; // Query embedding  

         $\mathbf{m}_{\text{key}} = \text{KeyNetwork}(M_q)$ ; // Memory keys  

         $\mathbf{m}_{\text{value}} = \text{ValueNetwork}(M_q)$ ; // Memory values  

         $\mathbf{A} = \frac{\mathbf{q}_{\text{query}} \cdot \mathbf{m}_{\text{key}}^T}{\alpha}$ ; // Attention scores  

         $\mathbf{P} = \text{Softmax}(\mathbf{A})$ ; // Attention probabilities  

         $\mathbf{m}_{\text{weighted}} = \mathbf{P} \cdot \mathbf{m}_{\text{value}}$ ; // Weighted memories  

         $\mathbf{m}_r = \text{Top-K}(\mathbf{P}, k)$ ; // Retrieved memories  

        Calculate the  $\text{Score}_{\text{LLM}}$  by using the LLM to  

        compare  $(\mathbf{m}_r, m'_r)$ , considering the query  $q$   

        and the current agent state.  

        Compute the Loss using  $\text{Score}_{\text{LLM}}$  as input to  

        the loss function.  

        Backpropagate(Loss)  

        Update model parameters  

    end  

    if validation performance improves then  

        // Save the current model state  

        SaveCheckpoint()  

    end  

end  

return Trained model

```

Algorithm 1. Training of the cross attention network for memory retrieval.

To further refine our model's memory retrieval capabilities, the cross attention weights across different memories in the memory bank are ranked, and the top- k memories are selected as the output m_r for the model's retrieved memory.

To ensure that retrieved memories effectively guide an agent's behavior during training, it is essential to assess their quality. While human evaluation is often considered the gold standard, it can be impractical due to its time-consuming nature, higher costs, and potential variability among evaluators. Recent studies have demonstrated that large language models (LLMs) can serve as reliable evaluators for natural language generation (NLG) tasks, exhibiting strong correlation with human judgments. For instance, Wang et al. (2023) found that ChatGPT achieved state-of-the-art or competitive correlation with human evaluations across various NLG tasks. Similarly, Chiang and Lee (2023) showed that LLM-based evaluations were consistent with expert human assessments in tasks such as open-ended story generation and adversarial attacks. Given the dynamic nature of agent interactions, LLMs offer

a consistent and scalable method for evaluating memory relevance, effectively considering context and quality.

Therefore, we employ an LLM to compare and score the memories retrieved by our model, denoted as m_r , against those retrieved using a baseline method, denoted as m'_r . This comparison is based on the current state of the agent and the memory query state, denoted by q . The LLM evaluates the relevance of m_r and m'_r to the agent's current state and query q , assigning scores based on their contextual appropriateness and alignment with the agent's goals on a scale from 1 to 10, producing scores $\text{Score}_{\text{LLM}}$ and $\text{Score}'_{\text{LLM}}$, respectively. The following loss function is then computed to train the model effectively:

$$\text{output_score} = \frac{\text{Score}_{\text{LLM}} - \text{Score}'_{\text{LLM}}}{10} \quad (4)$$

$$\text{loss} = \max(-\log(\text{output_score} + 1), 0) \quad (5)$$

The cross attention mechanism within our model dynamically ranks and retrieves memories based on their relevance to the given query q , leveraging the current agent state for context. This process not only enhances the responsiveness of the model to the evolving scenario within the agent environment but also aligns the retrieved memories more closely with the needs of the agent.

The loss function of our model is meticulously designed to optimize memory retrieval capabilities. It is defined as the logarithm of the normalized difference between scores assigned to the model-generated and baseline memories, effectively penalizing deviations from expected outcomes. This approach ensures the model not only learns to accurately retrieve relevant memories but also continually refines its retrieval process based on ground truth data, enhancing its adaptability in real-world scenarios.

To support this advanced training approach, the model parameters are finely tuned using the Adam optimizer. This optimizer is chosen for its ability to efficiently manage sparse gradients and adaptively adjust learning rates, which are vital for quickly converging to the most effective solutions.

The integration of a cross attention network, optimized through the use of large language models, further enhances the model's memory retrieval capabilities. This setup improves the efficiency and relevance of how memories are accessed within generative agents, leveraging the computational power of LLMs to refine the training process effectively. The use of LLMs to guide the training process allows our model to operate effectively with the support of advanced AI technologies, thereby making a significant contribution to the field of AI-driven memory management.

4 Results

4.1 Result analysis of auxiliary cross attention network

For the generation of our training dataset, we simulated the behavior of a pre-defined community of eight agents over three consecutive days, each consisting of 16 h of interactions. During these simulations, agents engaged in various tasks, similar to the agent-based interactions described in Generative Agents (Park

et al., 2023). Each agent's behavior was guided by ChatGPT (GPT-3.5-turbo), which generated context-specific interactions and stored the outcomes as memories. At every decision-making step, the agents' current state, past memories, and retrieved memories [ranked by the Weighted Memory Retrieval (WMR) method] were saved in the memory bank and vectorized using the text-embedding-ada-002 model, producing an embedding size of 1,536. Each training data entry included the agent's current state q , the corresponding memory bank M_q , and the WMR retrieved memories m' , ranked based on Recency, Importance, and Relevance.

The structure of a single training entry consisted of the agent's query (current state), the action taken, the type of action (e.g., interaction, decision), the prompt guiding the action, and the retrieved memories at that point in time. This complete data structure captures how an agent's decision is informed by both past experiences and context-specific information, ensuring a comprehensive training process. In total, 1,280 unique training entries were generated, each encapsulating the dynamic interaction between the agents and their environments, enhancing data diversity and robustness.

Once the training dataset was prepared, we configured the training parameters for the memory retrieval model. We used the Adam optimizer with a learning rate of 0.001 and a batch size of 16, while the text embeddings for memory were fixed at a size of 1,536. During retrieval, the model output the top five memories ranked by attention weights. The entire training process was executed on an NVIDIA RTX 4060 GPU, which significantly accelerated the model's convergence. Each agent's interaction data, including the current state, query, and retrieved memories, were incorporated into the model to optimize the memory retrieval process for generative agents in multi-agent settings.

The effectiveness of the model's training under the assistance of a LLM is demonstrated in Figure 3. This figure illustrates the significant decrease in training loss across epochs.

As illustrated in Figure 3, the model demonstrates significant improvement under the guidance of LLMs. The training loss declines sharply from an initial value of 1.5 to 0.12. This reduction is driven by the loss function, which incorporates scores provided by the LLM based on the agent's current context, to assess the memories retrieved by both the proposed and baseline methods. This downward trend indicates the model's increasing effectiveness in adapting to the data, optimizing parameter adjustments to better capture and utilize representative memories. Consequently, this enhancement enables the model to consistently outperform the baseline method in memory scoring, contributing to the significant reduction in loss.

To rigorously evaluate the performance of our proposed model, we conducted test simulations spanning a complete day, covering 16 h, using both the WMR memory retrieval method and the ACAN memory retrieval method based on the fully trained Auxiliary Cross Attention Network. The test involved eight agents, each representing different professions and personalities, consistent with the setup used during the training phase. These simulations generated a total of 435 data entries for comparative analysis. Given the nascent stage of research in this area, particularly regarding LLM-based generative agents, the baseline memory retrieval method we used Park et al. (2023) represents one of the most state-of-the-art approaches currently available for

comparison in agent memory retrieval. This ensures a fair and meaningful benchmark against which the performance of our ACAN model could be evaluated.

The assessment of the test data was conducted in the same manner as during training, where a large language model was employed in conjunction with the agent's contextual state to score the memories generated during the simulation on a scale from 1 to 10. We compared the memory retrieval scores from the ACAN model with those retrieved using the WMR memory retrieval method across all test data.

As illustrated in Figure 4, the results of memory retrieval using the ACAN model in comparison with the WMR method show that the ACAN method consistently achieves higher memory scores than the baseline. Specifically, the ACAN group scored an average of 5.94 with a standard deviation of 1.66, whereas the baseline group scored an average of 5.05 with a standard deviation of 1.88. Statistical tests further validate the significance of these differences, with a T-statistic of 7.44 and a corresponding P-value of 2.42×10^{-13} , significantly below the common significance level of 0.05. This strongly indicates that the ACAN model substantially outperforms the baseline method in terms of memory retrieval effectiveness.

The superior performance of the ACAN model can be attributed to its dynamic cross-attention mechanism, which optimizes memory retrieval by continuously adapting to the agent's evolving state and context. This mechanism allows the ACAN model to rank memories based not only on basic relevance metrics such as recency but also on a more nuanced evaluation of the importance of past experiences, as influenced by real-time feedback from the LLM. In contrast, the WMR method relies on static retrieval strategies that do not account for these contextual factors, leading to less accurate and less relevant memory retrieval.

Furthermore, the reduced standard deviation in the ACAN results indicates that the model consistently performs well across different scenarios, demonstrating its robustness in diverse environments. The WMR method, with a higher standard deviation, shows more variability in its effectiveness, suggesting that its performance is more dependent on specific scenarios or task conditions.

These findings also have broader implications for agent behavior and decision-making. By retrieving more relevant and contextually appropriate memories, the ACAN model enhances the agent's ability to make informed decisions that closely mimic human-like responses. This, in turn, improves the quality of the agent's interactions with both the environment and other agents. The results provide empirical support for the hypothesis that the ACAN model's memory retrieval mechanism leads to more natural and effective decision-making processes in multi-agent settings.

In addition, a deeper analysis of the memory scores reveals that the ACAN model particularly excels in scenarios that require the integration of complex, long-term memories. This suggests that the model's cross-attention mechanism not only improves short-term relevance but also facilitates the retrieval of critical long-term memories that might otherwise be overlooked in traditional retrieval methods. This highlights the potential for the ACAN model to enhance not only immediate decision-making but also more complex tasks involving strategic planning and social interactions.

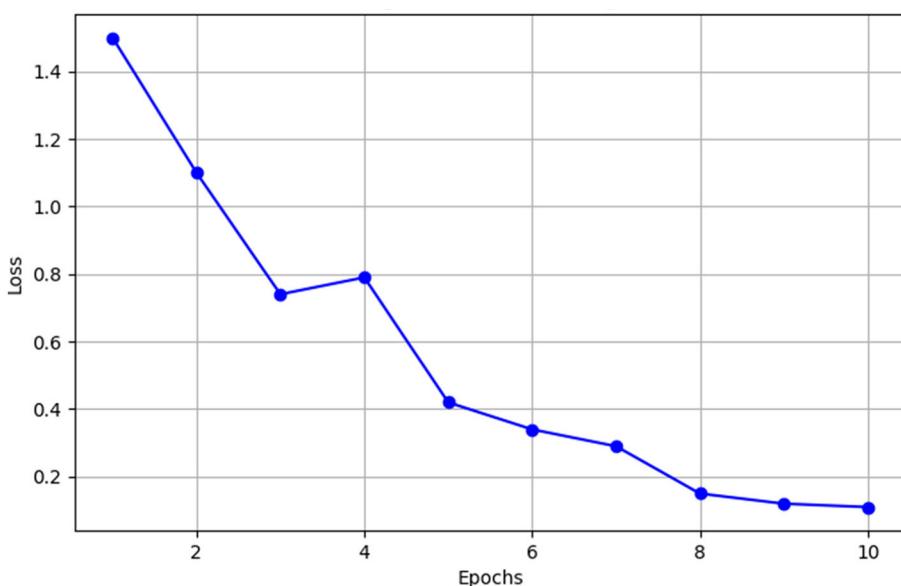


FIGURE 3
Training loss curve over epochs.

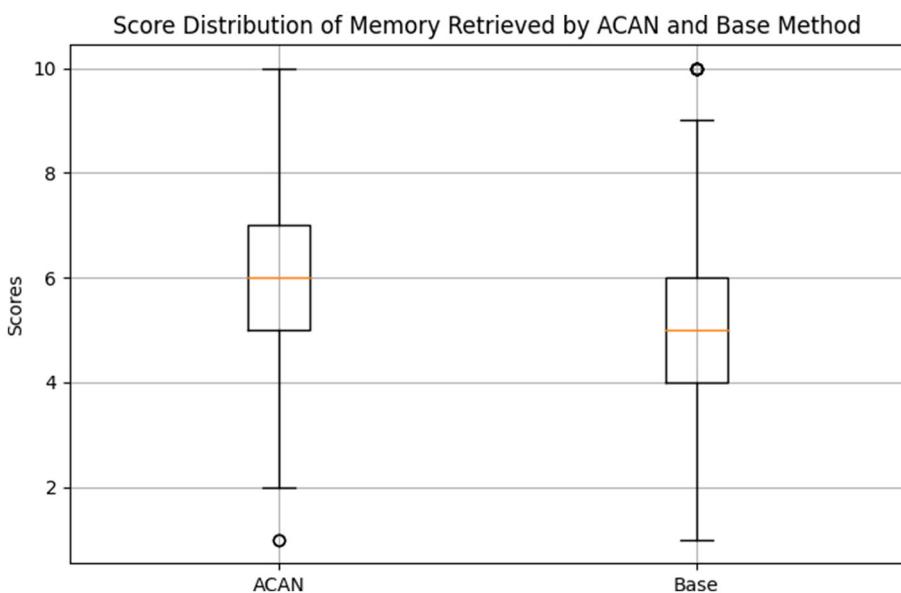


FIGURE 4
Score distribution of memory retrieved by ACAN and weighted memory retrieval (WMR).

4.2 Quantitative analysis of memory retrieval

While the training of our model and the assessments on the generated test set were conducted with the support of LLMs, we aimed to analyze the effectiveness of our proposed memory retrieval method without direct LLM intervention. For this purpose, a quantitative analysis experiment was designed, where agents received specific invitations under different memory retrieval modes, and their attendance probabilities

were compared. This experimental setup allows us to evaluate how different memory retrieval strategies impact the agents' perception of external stimuli and their cognitive ability to mimic human behavior.

For the experimental design, each day, random agents were invited at a specified hour to attend events at designated locations, occurring for 10 h excluding sleep times. Following the agents' agreement to attend, their actual appearance at the event at the appointed time was recorded. This simulation was conducted over a span of ten days, involving eight agents under two

TABLE 1 Detailed attendance rates across five trials for ACAN and weighted memory retrieval (WMR).

Metric	ACAN	WMR
Trial 1	35%	27%
Trial 2	36%	29%
Trial 3	29%	19%
Trial 4	32%	23%
Trial 5	31%	25%
Mean attendance rate	32.6%	24.6%
Standard deviation	2.881%	3.847%

different memory modes, generating a total of 100 invitation and attendance records.

As shown in Table 1, we compared the memory retrieval effectiveness of the ACAN model and the WMR method across five trials. In these experiments, agents using the ACAN model adhered to invitations 32.6% of the time on average, whereas agents using the WMR method adhered 24.6% of the time. This indicates that agents employing the ACAN model have a significantly higher likelihood of attending the events, with an average attendance rate that is eight percentage points higher than that of the WMR method. Additionally, observing the memory retrieval process of the agents revealed that those who successfully attended the events could accurately recall the relevant invitation information, further validating the ACAN model's effectiveness in enhancing memory retrieval accuracy.

To further quantify the statistical significance of this difference, a paired samples *t*-test was conducted. The *t*-test results yielded a T-statistic of 11.31 and a *P*-value of 0.00035, indicating that the observed difference in attendance rates between the two methods is highly significant (well below the common significance threshold of 0.05). This provides strong evidence that the ACAN model substantially improves the agents' responsiveness to invitations and their likelihood of attending events compared to the baseline WMR method.

Additionally, the standard deviations across the five trials show some variability in the results (2.881% for the ACAN method and 3.847% for the baseline method), but the ACAN model consistently outperformed the baseline, demonstrating both its stability and reliability. These findings emphasize the robustness of the ACAN-based memory retrieval approach in enhancing agents' event attendance behavior and improving their ability to respond to interactions within dynamic and complex simulated environments.

These findings demonstrate the robustness of the ACAN-based memory retrieval approach in enhancing agents' event attendance behavior and their responsiveness to interactions within dynamic and complex simulated environments. The integration of cross-attention mechanisms in ACAN likely facilitates better contextual understanding and memory utilization, which in turn leads to more effective decision-making and engagement in scheduled events. Consequently, the higher attendance rates associated with the ACAN model not only reinforce its effectiveness but also highlight its potential to simulate complex human-like social behaviors. This makes ACAN a valuable tool for applications that

require nuanced, contextually-aware decision-making, enhancing the capability of agents to navigate and adapt within multifaceted interactive settings.

5 Discussion

This study successfully developed and implemented a text-based generative agent simulation world, creating a community with multiple locations and agents that engage in various interactions. Based on this foundation, we designed an innovative memory retrieval system using the Auxiliary Cross Attention Network. This system simulates human behavior by ranking the attention weights between the agent's current state and memories in the memory bank, retrieving the memories most relevant to the current state. To train this model, we introduced an innovative approach by leveraging the assistance of LLMs. During training, the LLM scores the memories retrieved by our model against those retrieved by the baseline method, using these scores along with a novel loss function to train the model effectively.

Our evaluations leveraged a test data set generated from simulations of LLM-based agent interactions, representing a typical day in the life of these agents. This simulated environment, along with our specially designed agent invitation and attendance experiments, provided a robust framework for validating the advantages of our memory retrieval method over traditional approaches. The results from these evaluations confirm that our system significantly enhances the memory retrieval process, thereby supporting more effective decision-making in generative agents. By optimizing how memories are retrieved and utilized, our method allows agents to respond in ways that are more closely aligned with human behavior based on their current state, thereby enriching their ability to engage in and navigate complex interpersonal interactions.

Despite the achievements of our study, there are notable limitations to consider. The model's effectiveness relies heavily on continuous evaluations by Large Language Models (LLMs), increasing computational demands and operational costs due to LLM API token usage. Additionally, LLM feedback slows training, potentially limiting rapid development and scalability. Our evaluation method, dependent on LLMs, may not generalize across different configurations or domains, and using LLMs instead of human assessment for training and testing could affect result rigor and objectivity, raising concerns about robustness and generalizability. However, recent work such as Edge et al. (2024) demonstrates that LLMs can reliably evaluate relevance and faithfulness in RAG systems, supporting their use as cost-effective alternatives to human assessments. To further enhance robustness, especially in nuanced scenarios, incorporating human validation may serve as a valuable complement.

The broader impacts of our Auxiliary Cross Attention Network (ACAN) model extend significantly across the AI discipline, introducing a novel adaptive framework for memory retrieval that not only enriches theoretical models of AI agent interactions but also demonstrates substantial practical applications. Leveraging LLM assessments to shape the loss function during training is an innovative approach that significantly refines the precision of memory retrieval. This advancement holds great promise

for revolutionizing human-agent interactions by enabling more natural and complex interpersonal simulations. Future research should focus on further enhancing the model's capabilities through more sophisticated neural network architectures which could advance the state of memory retrieval in AI agents. Additionally, developing autonomous feedback mechanisms will be crucial for advancing AI agents that can adapt independently to dynamic environments, thus pushing the boundaries of what is possible in Artificial General Intelligence. This focus on improving memory retrieval systems directly supports the evolution of more intelligent and responsive AI agents, paving the way for broader and more effective implementations in various AI-based domains.

6 Conclusions

This study has introduced the Auxiliary Cross Attention Network (ACAN), a pioneering memory retrieval system for generative agents, showcasing a significant advancement in AI agent driven by large language models (LLMs). ACAN effectively enhances agent adaptability and behavioral consistency by dynamically ranking and retrieving memories based on the agent's current state, thus addressing the critical need for sophisticated memory management mechanisms in Artificial General Intelligence. While the reliance on LLMs for training and evaluating the system poses challenges for scalability and efficiency, it simultaneously highlights the need for innovations that could decrease such dependencies and enhance the autonomy of the system. This research not only demonstrates the potential of ACAN in improving memory retrieval within varied agent interactions but also highlights the broader applicability of LLMs in advancing AI technologies. Moving forward, the focus will be on refining these methodologies to further enhance the capabilities and independence of AI agents in complex environments.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found at: <https://github.com/HongChuanYang/Training-by-LLM-Enhanced-Memory-Retrieval-for-Generative-Agents-via-ACAN>.

References

- Abdalla, R., and Mishra, A. (2021). Agent-oriented software engineering methodologies: analysis and future directions. *Complexity* 2021:1629419. doi: 10.1155/2021/1629419
- Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F., et al. (2023). GPT-4 technical report. *arXiv preprint arXiv:2303.08774*. doi: 10.48550/arXiv.2303.08774
- Alonso, N., Figoliola, T., Ndirango, A., and Millidge, B. (2024). Toward conversational agents with context and time sensitive long-term memory. *arXiv preprint arXiv:2406.00057*. doi: 10.48550/arXiv.2406.00057
- Bang, Y., Cahyawijaya, S., Lee, N., Dai, W., Su, D., Wilie, B., et al. (2023). A multitask, multilingual, multimodal evaluation of chatGPT on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*. doi: 10.18653/v1/2023.ijcnlp-main.45
- Bates, J. (1994). The role of emotion in believable agents. *Commun. ACM* 37, 122–125. doi: 10.1145/176789.176803
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., et al. (2020). Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* 33, 1877–1901. doi: 10.48550/arXiv.2005.14165
- Bulatov, A., Kuratov, Y., and Burtsev, M. (2022). Recurrent memory transformer. *Adv. Neural Inf. Process. Syst.* 35, 11079–11091. doi: 10.5555/3600270.3601075
- Burtsev, M. S., Kuratov, Y., Peganova, A., and Sapunov, G. V. (2020). Memory transformer. *arXiv preprint arXiv:2006.11527*. doi: 10.48550/arXiv.2006.11527
- Callison-Burch, C., Tomar, G. S., Martin, L. J., Ippolito, D., Bailis, S., and Reitter, D. (2022). Dungeons and dragons as a dialog challenge for artificial intelligence. *arXiv preprint arXiv:2210.07109*. doi: 10.18653/v1/2022.emnlp-main.637

Author contributions

CH: Conceptualization, Investigation, Methodology, Software, Writing – original draft. QH: Data curation, Visualization, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported by the talent introduction project of the School of Finance and Economics at Anhui Science and Technology University, titled “Research on Data-Driven Chance Constraint Optimization Problem with Application to Portfolio Management” (grant number CJYJ202401).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that Gen AI was used in the creation of this manuscript. During the preparation of this work, the author(s) used ChatGPT-4o to assist in refining grammatical accuracy in the writing process. After using this tool, the author(s) thoroughly reviewed and edited the content as needed and take full responsibility for the content of the publication.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Cao, Y., Li, S., Liu, Y., Yan, Z., Dai, Y., Yu, P. S., et al. (2023). A comprehensive survey of AI-generated content (AIGC): a history of generative ai from gan to chatGPT. *arXiv preprint arXiv:2303.04226*. doi: 10.48550/arXiv.2303.04226
- Chiang, C.-H., and Lee, H.-y. (2023). Can large language models be an alternative to human evaluations? *arXiv preprint arXiv:2305.01937*. doi: 10.48550/arXiv.2305.01937
- Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., et al. (2024). From local to global: a graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*. doi: 10.48550/arXiv.2404.16130
- Freiknecht, J., and Effelsberg, W. (2020). "Procedural generation of interactive stories using language models," in *Proceedings of the 15th International Conference on the Foundations of Digital Games* (New York, NY: Association for Computing Machinery), 1–8. doi: 10.1145/3402942.3409599
- Gronauer, S., and Diepold, K. (2022). Multi-agent deep reinforcement learning: a survey. *Artif. Intell. Rev.* 55, 895–943. doi: 10.1007/s10462-021-09996-w
- Guo, M., Ainslie, J., Uthus, D., Ontanon, S., Ni, J., Sung, Y.-H., et al. (2021). Long5: efficient text-to-text transformer for long sequences. *arXiv preprint arXiv:2112.07916*. doi: 10.18653/v1/2022.findings-naacl55
- Hou, Y., Tamoto, H., and Miyashita, H. (2024). "my agent understands me better: integrating dynamic human-like memory recall and consolidation in LLM-based agents," in *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 1–7. doi: 10.1145/3613905.3650839
- Hu, C., Fu, J., Du, C., Luo, S., Zhao, J., and Zhao, H. (2023). ChatDB: augmenting LLMS with databases as their symbolic memory. *arXiv preprint arXiv:2306.03901*. doi: 10.48550/arXiv.2306.03901
- Huang, Z., Gutierrez, S., Kamana, H., and MacNeil, S. (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* (New York, NY: Association for Computing Machinery), 1–3. doi: 10.1145/3586182.3615796
- Jiao, W., Wang, W., Huang, J.-t., Wang, X., and Tu, Z. (2023). Is chatGPT a good translator? A preliminary study. *arXiv preprint arXiv:2301.08745*. doi: 10.48550/arXiv.2301.08745
- Kagaya, T., Yuan, T. J., Lou, Y., Karlekar, J., Pranata, S., Kinose, A., et al. (2024). RAP: retrieval-augmented planning with contextual memory for multimodal LLM agents. *arXiv preprint arXiv:2402.03610*. doi: 10.48550/arXiv.2402.03610
- Khattab, O., Santhanam, K., Li, X. L., Hall, D., Liang, P., Potts, C., et al. (2022). Demonstrate-search-predict: composing retrieval and language models for knowledge-intensive NLP. *arXiv preprint arXiv:2212.14024*. doi: 10.48550/arXiv.2212.14024
- Laird, J., and VanLent, M. (2001). Human-level AI's killer application: interactive computer games. *AI Mag.* 22:15–15. doi: 10.1609/aimag.v22i2.1558
- Liang, X., Wang, B., Huang, H., Wu, S., Wu, P., Lu, L., et al. (2023). Unleashing infinite-length input capacity for large-scale language models with self-controlled memory system. *arXiv preprint arXiv:2304.13343*. doi: 10.48550/arXiv.2304.13343
- Lin, J., Zhao, H., Zhang, A., Wu, Y., Ping, H., and Chen, Q. (2023). Agentsims: an open-source sandbox for large language model evaluation. *arXiv preprint arXiv:2308.04026*. doi: 10.48550/arXiv.2308.04026
- Liu, R., Yang, R., Jia, C., Zhang, G., Zhou, D., Dai, A. M., et al. (2023). Training socially aligned language models in simulated human society. *arXiv preprint arXiv:2305.16960*. doi: 10.48550/arXiv.2305.16960
- Modarressi, A., Imani, A., Fayyaz, M., and Schütze, H. (2023). Ret-LLM: towards a general read-write memory for large language models. *arXiv preprint arXiv:2305.14322*. doi: 10.48550/arXiv.2305.14322
- Mohtashami, A., and Jaggi, M. (2023). Random-access infinite context length for transformers. *Adv. Neural Inf. Process. Syst.* 36, 54567–54585. doi: 10.48550/arXiv.2305.16300
- Musacchio, E., Siciliani, L., Basile, P., and Semeraro, G. (2024). Adapting large language models to narrative content. *Complexity* 6:8.
- Neelakantan, A., Xu, T., Puri, R., Radford, A., Han, J. M., Tworek, J., et al. (2022). Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*. doi: 10.48550/arXiv.2201.10005
- Nuxoll, A. M., and Laird, J. E. (2007). "Extending cognitive architecture with episodic memory," in *AAAI* (Palo Alto, CA: AAAI Press), 1560–1564.
- OpenAI. (2022). *Introducing ChatGPT*. Available online at: <https://openai.com/blog/chatgpt> (accessed April 3, 2023).
- Park, H. H., Vyas, Y., and Shah, K. (2022). Efficient classification of long documents using transformers. *arXiv preprint arXiv:2203.11258*. doi: 10.18653/v1/2022.acl-short.79
- Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. (2023). "Generative agents: interactive simulacra of human behavior," in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (New York, NY: Association for Computing Machinery), 1–22. doi: 10.1145/3586183.3606763
- Park, J. S., Popowski, L., Cai, C., Morris, M. R., Liang, P., and Bernstein, M. S. (2022). "Social simulacra: creating populated prototypes for social computing systems," in *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (New York, NY: Association for Computing Machinery), 1–18. doi: 10.1145/3526113.3545616
- Pesce, E., and Montana, G. (2020). Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication. *Mach. Learn.* 109:1727–1747. doi: 10.1007/s10994-019-05864-5
- Qian, C., Cong, X., Yang, C., Chen, W., Su, Y., Xu, J., et al. (2023). Communicative agents for software development. *arXiv preprint arXiv:2307.07924*. doi: 10.48550/arXiv.2307.07924
- Russell, S. J., and Norvig, P. (2016). *Artificial Intelligence: A Modern Approach*. Upper Saddle River, NJ: Pearson.
- Schuurmans, D. (2023). Memory augmented large language models are computationally universal. *arXiv preprint arXiv:2301.04589*. doi: 10.48550/arXiv.2301.04589
- Schwabe, L., Nader, K., and Pruessner, J. C. (2014). Reconsolidation of human memory: brain mechanisms and clinical relevance. *Biol. Psychiatry* 76:274–280. doi: 10.1016/j.biopsych.2014.03.008
- Schweitzer, F., Krivachy, T., and Garcia, D. (2020). An agent-based model of opinion polarization driven by emotions. *Complexity* 2020:5282035. doi: 10.31235/osf.io/8m2wq
- Serrano, S., and Smith, N. A. (2019). Is attention interpretable? *arXiv preprint arXiv:1906.03731*. doi: 10.48550/arXiv.1906.03731
- Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. (2023). Reflexion: language agents with verbal reinforcement learning. *Adv. Neural Inf. Process. Syst.* 36, 8634–8652. doi: 10.5555/3666122.3666499
- Squire, L. R. (1986). Mechanisms of memory. *Science* 232, 1612–1619. doi: 10.1126/science.3086978
- Sridharan, S., Stevens, J. R., Roy, K., and Raghunathan, A. (2023). X-former: in-memory acceleration of transformers. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 31, 1223–1233. doi: 10.1109/TVLSI.2023.3282046
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst.* 30, 6000–6010. doi: 10.5555/3295222.3295349
- Wang, J., Liang, Y., Meng, F., Sun, Z., Shi, H., Li, Z., et al. (2023). Is chatGPT a good nlg evaluator? A preliminary study. *arXiv preprint arXiv:2303.04048*. doi: 10.18653/v1/2023.newsuum-1.1
- Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., et al. (2024). A survey on large language model based autonomous agents. *Front. Comput. Sci.* 18:186345. doi: 10.1007/s11704-024-40231-1
- Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., et al. (2022). Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*. doi: 10.48550/arXiv.2206.07682
- Wooldridge, M., and Jennings, N. R. (1995). Intelligent agents: theory and practice. *Knowl. Eng. Rev.* 10:115–152. doi: 10.1007/3-540-58855-8
- Wu, T., Jiang, E., Donsbach, A., Gray, J., Molina, A., Terry, M., et al. (2022). "Promptchainer: chaining large language model prompts through visual programming," in *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, 1–10. doi: 10.1145/3491101.3519729
- Wu, Y., Rabe, M. N., Hutchins, D., and Szegedy, C. (2022). Memorizing transformers. *arXiv preprint arXiv:2203.08913*.
- Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., et al. (2023). The rise and potential of large language model based agents: a survey. *arXiv preprint arXiv:2309.07864*. doi: 10.48550/arXiv.2309.07864
- Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., et al. (2024). Harnessing the power of llms in practice: a survey on chatGPT and beyond. *ACM Trans. Knowl. Discov. Data.* 18, 1–32. doi: 10.1145/3649506
- Yannakakis, G. N. (2012). "Game AI revisited," in *Proceedings of the 9th Conference on Computing Frontiers*, 285–292. doi: 10.1145/2212908.2212954
- Yu, Y., Li, H., Chen, Z., Jiang, Y., Li, Y., Zhang, D., et al. (2024). Finmem: a performance-enhanced llm trading agent with layered memory and character design. *Proc. AAAI Symp. Ser.* 3, 595–597. doi: 10.1609/aaaiiss.v3i1.31290
- Zalta, E. N., Nodelman, U., and Allen, C. (1995). *Stanford Encyclopedia of Philosophy*. Stanford, CA: The Metaphysics Research Lab, Philosophy Department, Stanford University.
- Zhang, Z., Bo, X., Ma, C., Li, R., Chen, X., Dai, Q., et al. (2024). A survey on the memory mechanism of large language model based agents. *arXiv preprint arXiv:2404.13501*. doi: 10.48550/arXiv.2404.13501
- Zhao, A., Huang, D., Xu, Q., Lin, M., Liu, Y.-J., and Huang, G. (2024). Expel: LLM agents are experiential learners. *Proc. AAAI Conf. Artif. Intell.* 38, 19632–19642. doi: 10.1609/aaai.v38i17.29936
- Zhong, W., Guo, L., Gao, Q., Ye, H., and Wang, Y. (2024). Memorybank: enhancing large language models with long-term memory. *Proc. AAAI Conf. Artif. Intell.* 38, 19724–19731. doi: 10.1609/aaai.v38i17.29946
- Zhou, X., Li, G., and Liu, Z. (2023). LLM as DBA. *arXiv preprint arXiv:2308.05481*. doi: 10.48550/arXiv.2308.05481