Received 24 March 2024, accepted 16 April 2024, date of publication 25 April 2024, date of current version 8 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3393485

RESEARCH ARTICLE

MemoryRepository for AI NPC

SHIJIE ZHENG[1], (Student Member, IEEE), KEITH HE[2], LE YANG[3], AND JIE XIONG[1]
[1]School of Electronic Information and Electrical Engineering, Yangtze University, Jingzhou 434023, China
[2]OgCloud Ltd., Guangzhou 510000, China
[3]School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China

Corresponding author: Jie Xiong (xiongjie@yangtzeu.edu.cn)

ABSTRACT Since the release of ChatGPT, large language models (LLMs) have played a huge role in various industries. In the field of games, we have used LLMs to act as intelligent AI NPC, which makes NPCs more intelligent. However, there is still an obvious obstacle -the LLMs lacks long-term memory and human-like memory mechanism. This flawed memory mechanism prevents NPCs from Long-term interaction and humanized memory based on conversation records. Recognizing the necessity of long-term memory and humanized memory, we proposed MemoryRepository, a memory mechanism for LLMs specifically used in the AI NPC field.MemoryRepository enables the model to have short-term memory and long-term memory. Short-term memory is more detailed and full, while long-term memory are more concise and partial. MemoryRepository is inspired by human memory and forgetting mechanisms. This mechanism allows AI NPCs to forget and summarize past conversation records, thereby providing long-term interaction capabilities. More importantly, this process of forgetting and summarizing the details of short-term memory into general long-term memories makes NPCs more human-like. MemoryRepository is versatile and can adapt to closed source models such as ChatGPT and open source models such as ChatGLM. To Intuitively verify the effectiveness of MemoryRepository in the field of AI NPC, we created an example in which all NPCs are represented by LLMs adapted to MemoryRepository. The example shows that by embedding LLM in MemoryRepository and fine-tuning NPCs character dialogue data, AI NPC can conduct better long-term conversations and appear more human-like during the interaction process. To validate the effectiveness of MemoryRepository, one hundred pieces of NPCs dialogue data were created and then quantitatively analyzed through evaluation indicators. The analysis results show that NPCs equipped with MemoryRepository can summarize and forget past memories, which enables it to have the ability to hold long-term conversations and conduct more human-like conversations.

INDEX TERMS AI NPC, human-like, long-term memory, MemoryRepository, LLM.

I. INTRODUCTION

In the realm of modern games, Non-Player Characters (NPCs) have emerged as critical components, assuming roles as adversaries, allies, and impartial entities that significantly contribute to the intricacies of game dynamics [1]. Despite remarkable advancements in visual and auditory technologies [2], the fundamental characteristics of NPCs have seen little evolution over the past thirty years. Their constrained ability for authentic understanding and interaction culminates in monotonous and predictable actions [3]. Conventionally, NPCs are bound to predetermined scripts, lacking the capability to engage in substantive dialogue [4]. Their responses remain uniform, disregarding the diversity of player inputs. The rigid nature of traditional interactions between NPCs and players often leads to a monotonous gaming experience. This monotony can significantly reduce player engagement and satisfaction. To enhance the gaming experience, it's essential to integrate NPCs with capabilities for sustained interactions and a memory system that more closely resembles human-like recall [5].

To address these limitations in NPCs interaction, game developers have traditionally relied on methods such as scripting and behavior trees. These techniques have been central to simulating an agent's memory and creating an illusion of forgetfulness. Originating in the early stages of video

The associate editor coordinating the review of this manuscript and approving it for publication was Anandakumar Haldorai.

game development, scripting was first used to direct NPCs behavior. Visionaries like John McCarthy, who played a crucial role in the development of AI scripting languages, laid the foundation for this approach [6]. As gaming complexity escalated, behavior trees, introduced by figures like Damian Isla in the context of 'Halo 2', became the preferred method, offering enhanced flexibility and decision-making processes for NPCs [7]. This transition marked a notable advancement in the NPCs interaction paradigm, enabling more nuanced and seemingly dynamic interactions.

Despite these developments, both scripting and behavior trees require substantial designer intervention to attain realism, a constant since their introduction. This reliance on intensive designer input underscores the limitations of these methods in achieving truly lifelike, adaptive NPCs behavior, thus underscoring the ongoing quest for more advanced solutions in modern game design [8]. The integration of LLMs into virtual agent development marks a significant leap forward in enhancing the authenticity of NPCs. This innovative approach deviates from traditional methods, leveraging the advanced capabilities of LLMs, which were significantly progressed by visionaries like Zhao et al. in their foundational work on neural network-based language models [9]. Gallotta et al. in "Large Language Models and Games: A Survey and Roadmap" demonstrated how LLMs can generate dynamic and contextually rich dialogues for NPCs [10]. Their research showed that LLMs, specifically fine-tuned for game narrative datasets, could produce dialogues that adapt to the player's actions and choices, significantly enhancing the narrative depth and player engagement. Zhou et al. in "Dialogue Shaping: Empowering Agents through NPC Interaction" explored the impact of LLM integration on the realism of NPC interactions [11]. They found that NPCs powered by LLMs exhibit more natural and believable behaviors, leading to a more immersive gaming experience. Their work underscored the potential of LLMs to transform static NPCs into dynamic entities that can evolve based on in-game events and player interactions. These studies collectively highlight the rapid advancements in applying LLMs to NPC development. By enabling more sophisticated dialogue generation, enhancing realism, and incorporating long-term memory, LLMs are redefining the possibilities for NPC interactions in video games, contributing to the creation of more engaging and lifelike virtual worlds. LLMs enable NPCs to engage in conversational interactions that are not confined to a predetermined response library but are driven by the model's sophisticated AI. This advancement allows the application of LLMs in NPCs development, a concept that has been explored in recent studies [12]. The superior language processing prowess of LLMs empowers NPCs to produce unique, context-sensitive responses. This capability represents a departure from the predictability of traditional scripted interactions, as noted in recent research [13], thereby enhancing player engagement and immersion.

However, a significant limitation of LLMs, as identified in studies by researchers like Panwar [14], is their constrained long-term memory. Although NPCs lacking long-term memory can engage in intelligent conversations through LLMs, their dialogues lack coherence. These NPCs can generate responses with AI-driven intelligence, but their inability to recall previous interactions diminishes the user experience. This deficiency impacts their ability to maintain coherent storylines over extended periods, presenting challenges in crafting evolving NPCs narratives. This means that most games that require consistent interaction between NPCs and players require NPCs to have long-term memory, especially games with a role-playing nature like Stardew Valley.

While the implementation of LLMs in NPCs has enhanced their interactivity, existing LLMs often lack long-term interaction and human-like interaction, crucial for realistic NPCs interactions [15]. To solve this problem, we developed MemoryRepository, an innovative memory mechanism specially designed for LLMs in the field of AI NPC. This mechanism mimics the human memory process, allowing AI NPC to forget short-term memory, thereby freeing up memory space [16]. Importantly, before this forgetting process, it summarizes these memories to retain the essential information in a more compact form. In some fields, it is inappropriate to sacrifice some short-term memory details to provide long-term memory. For the NPC field, this gradual forgetting of short-term memory details and the gradual accumulation of long-term memory make our NPCs more human-like. This method is conducive to the effective use of NPCs's short-term and long-term memory capabilities. Through MemoryRepository, NPCs can forget and succinctly retell past events, thus promoting dynamic and long-lasting interactions. The MemoryRepository mechanism we have developed enhances the interactive capabilities of NPCs, enabling them to demonstrate a nuanced understanding of historical events and player dynamics, and possess stronger human-like interactions. This sophisticated functionality is pivotal for crafting complex storylines and propelling dynamic character evolution, thereby augmenting the narrative's richness and the immersive quality of the gaming experience. Our strategy not only elevates the realism and engagement in NPC interactions but also sets the stage for pioneering narrative and gameplay paradigms, ultimately delivering a deeply enriched and satisfying player experience.

In our research, we developed a game case featuring AI NPC based on LLMs integrated with MemoryRepository. This game case demonstrates the practical significance of MemoryRepository in several ways. Firstly, its adaptability to various situations is evidenced by the fine-tuning of 50k dialogues from game characters, enhancing its effectiveness as an NPCs [17]. Additionally, it employs both the open-source ChatGLM and the closed-source ChatGPT models, showcasing MemoryRepository's adaptability across different LLM platforms. To evaluate the effectiveness of MemoryRepository, we conducted both qualitative and quantitative analyses.

The qualitative evaluation involved analyzing real-world user conversations, comparing common LLMs with and without the integration of MemoryRepository. For quantitative analysis, we employed simulated interactions. An in-memory store of conversations was created, spanning ten days and involving diverse virtual user personas, each characterized with unique personalities as portrayed by ChatGPT [18]. This approach allowed us to pose 400 exploratory questions to assess the model's memory recall and response relevance. Furthermore, we conducted a case study to empirically illustrate the enhancements MemoryRepository brings to AI NPC in terms of human-like interactions and long-term engagement effects. The results from this comprehensive analysis unequivocally indicate that NPCs utilizing our game case study excel in long-term interaction and human-like engagement, affirming MemoryRepository's potential to significantly enhance LLM performance in scenarios involving prolonged interactions. In conclusion, this paper presents the following key contributions:

- We introduced MemoryRepository, a human-like long-term memory mechanism created for AI NPC. It enables LLMs to forget and summarize, thus having short-term and long-term memory similar to real humans, which makes them more human-like.
- We created a sample, A game with AI NPC powered by MemoryRepository and fine-tuned based on game NPCs. NPCs in the game can have long-term interactions and human-like conversations.
- We demonstrate the enhancements of MemoryRepository in the AI NPC field from four key aspects: (1) Memory retrieval accuracy; (2) Human-like Interaction;(3) Contextual coherence;(4) Answer correctness;

II. RELATED WORKS

The evolution of LLMs, exemplified by OpenAI's GPT series, has significantly impacted natural language processing. The introduction of GPT-3 Brown et al. and its successor [19], GPT-4 (OpenAI, 2023), marked substantial advancements in language understanding and generation [20]. The practical application of LLMs in NPCs development has been validated by works such as 'Generative Agents: Interactive Simulacra of Human Behavior' Park et al. [7], showcasing the feasibility of LLMs in this domain. However, this research also points out areas needing improvement, especially in long-term interaction and human-like qualities of NPCs. Building on this foundation, our research aims to address these gaps, enhancing NPCs interactions in virtual environments.

As LLMs continue to evolve, substantial research efforts have been dedicated to enhancing neural models with advanced memory capabilities. A significant development in this domain is the progress in Memory-Augmented Neural Networks (MANNs), which address traditional neural networks' limitations in handling complex mappings and relearning from fresh data [21], [22]. A key advancement in this field, as detailed by Geethan Karunaratne et al. in Nature Communications (2021), is the introduction of a robust architecture utilizing computational memory units for high-dimensional vector processing. This approach employs a content-based attention mechanism for efficient memory processing. Moreover, the unique architecture of MANNs, integrating a neural network controller with structured explicit memory, facilitates rapid assimilation of new concepts without overwriting previously learned information [23]. The controller's interaction with content-addressable memory further highlights the innovative strides in memory augmentation for neural networks [24]. Another notable recent development in the field is MemGPT: Towards LLMs as Operating Systems Packer et al. [25], which introduces virtual context management inspired by hierarchical memory systems in traditional operating systems. This technique enables MemGPT to effectively extend the context within LLMs' restricted context windows. By intelligently managing various memory tiers and utilizing interrupts, MemGPT enhances long-term interaction capabilities [25]. These developments mark a significant leap in enhancing neural models' memory capabilities, paving the way for more complex and efficient AI systems. For MANNs, despite the progress in the field of LLMs, their long-term interaction capacity is constrained by memory limitations, which hinder sustained interactions and the input of sufficient preset information to achieve interactions more akin to human-like ones [26], [27]. MemGPT addresses some aspects of long-term interaction but lacks truly human-like long-term interaction. Unlike other domains that require long-term interaction, in the AI NPC field, our demand extends beyond mere long-term interaction; What we really pursue is human-like long-term interaction. Integrating long-term and human-like interaction capabilities into LLMs for AI NPC applications remains an unmet but critically essential need [28], [29]. Our MemoryRepository initiative aims to bridge this gap, proposing innovative solutions to enhance the personalization and continuity of interactions in LLMs. This approach not only resolves the existing memory function gap but also notably strengthens the human-like interaction capabilities of NPCs in the AI NPC domain. Importantly, our proposal does not introduce a new model, but rather a memory system mechanism that can be applied to existing models.

III. ARCHITECTURE OF MEMORYREPOSITORY

We propose MemoryRepository, a tailored memory mechanism designed specifically for LLMs to enhance AI NPC agents. In this section, we will detail the architecture of MemoryRepository. The scheduling policy governing MemoryRepository will be discussed in the following chapter. A comprehensive glossary of the terms used for modeling and formulating the data is presented in Table 1. Regarding the architecture of MemoryRepository, as illustrated in Figure 1, it contains three primary components.

- Memory Room: This serves as the memory storage area and is further divided into two sub-parts: the Short-Term MemoryRoom and the Long-Term MemoryRoom.

TABLE 1. Notation used for modeling and scheduling.

Parameter	Description
LLM	The LLMs(large language models) is employed within the testing environment of Platform P.
token	In the context of LLMs, A token refers to the basic unit of input and output used by the model,LLMs have an upper limit on the number of tokens they can process due to computational constraints.
T	Function to calculate token value
S	Function to summary memory
T_{limit}	The upper limit of the number of Tokens
Prompt	Prompt refers to a piece of text or question as input to the model. It is used to guide the model to generate text of a specific type or content
P	P is the abbreviation for prompt
Completions	Completions refer to the text or answers generated by the model. When you provide a prompt to the model, the model will generate one or more possible completions as an answer based on the prompt.
C	C is the abbreviation for Completions
M	Memory of LLMs
M_l	Long-term memory of LLMs, Long-term memory comes from forgetting and summarizing a series of short-term memories
M_s	Short-term memory of LLMs, Short-term memory is limited and is used to store the latest memory. The capacity is not large.
M_{meta}	Metamemory, as initialization memory, is used to control the fixed personality of AI NPC.
M_m	We define a summary set M_m for every m dialogue turns
W	Evaluation metrics for traceback completeness
E	Embedding, embedding refers to the process of mapping text or data into a low-dimensional vector space. Embedding is a representation that converts high-dimensional discrete data (such as words, sentences, or documents) into a continuous vector representation.

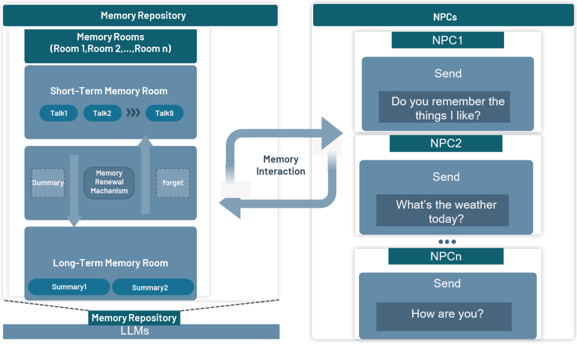


FIGURE 1. MemoryRepository Structure.

- Memory interaction: This component facilitates the retrieval and interaction with the stored memories in the MemoryRoom.
- Memory renewal mechanism: This mechanism is responsible for simulating human summarization and forgetting processes by periodically updating and refreshing the memories stored in the MemoryRoom.Composed of memory forgetting mechanism and memory enhancement mechanism.

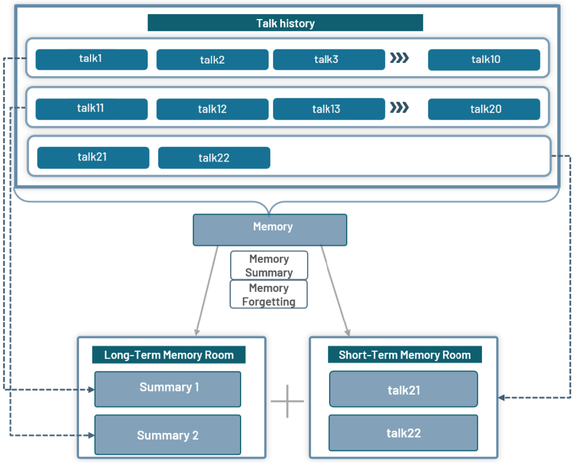


FIGURE 2. MemoryRepository Processing.

A. MEMORY ROOM

The MemoryRepository consists of numerous MemoryRooms, the number of which directly corresponds to the count of NPCs. Each NPC's memories are housed within their own dedicated MemoryRoom. Memory Room is similar to the human short-term memory and long-term memory systems. Through long-term memory, humans can store more information without changing the upper limit of memory. Given that the total memory capacity is limited, for each MemoryRoom, we strategically divide it into these two rooms.The Short-Term Memory Room is designated for storing recent information. This segment of memory, while limited in capacity and less durable, is rich in detail, mirroring the nature of human short-term memory in its precision and immediacy. On the other hand, the Long-Term Memory Room is reserved for storing more enduring memories. The details in this section are comparatively broader and less fine-grained, yet it benefits from a larger capacity and greater persistence, akin to the human brain's long-term memory storage. This division allows for an efficient and realistic simulation of memory processing in AI NPC agents, enhancing their interaction capabilities and realism.

1) SHORT-TERM MEMORY ROOM

Short-term memory is responsible for storing recent memories and has a limited memory capacity. It can only retain memories from the immediate past and cannot store distant memories. Each conversation is assigned an index, which creates an organized narrative of past interactions. This detailed record not only enables accurate memory retrieval but also facilitates the subsequent memory update process by providing a comprehensive index of the conversation history.

2) LONG-TERM MEMORY ROOM

Long-term memory plays a vital role in information retention by storing memories derived from previous short-term memory. As short-term memory gradually forgets and summarizes, long-term memory is formed, allowing for the preservation of important knowledge and experiences.

B. MEMORY INTERACTION

In our MemoryRepository framework, we've innovated NPC dialogue interactions by integrating both short-term and long-term memory. This integration is achieved through our novel Interactive Dual-Tower DSSM (ID-DSSM), which evolves the classic Deep Semantic Similarity Model (DSSM). The classical DSSM, while effective in semantic understanding, often struggles with dynamic context adaptation in ongoing dialogues. Our ID-DSSM addresses this limitation by introducing a dynamic interaction feature, which significantly refines memory retrieval, thus enhancing contextuality and accuracy.

Drawing inspiration from advancements in Dense Passage Retrieval, ID-DSSM employs a dual-tower architecture, renowned for its enhanced knowledge extraction capabilities. This architecture treats each dialogue exchange as an individual memory unit, which undergoes contextual encoding. This process transforms these units into a pre-processed set of memories that are efficiently retrievable.

To facilitate rapid access to these memory vectors, we utilize Approximate Nearest Neighbor Indexing (ANNI), a technique akin to FAISS (Facebook AI Similarity Search). FAISS is a library for efficient similarity search and clustering of dense vectors, enabling the quick retrieval of data from large datasets. In our model, ANNI aids in swiftly accessing these memory vectors. Concurrently, the current conversational context is vectorized, serving as a dynamic guide to identify the most relevant memory unit for any given dialogue instance.

The adaptability of the encoder within ID-DSSM is particularly notable. It can be tailored for a variety of conversational AI applications, making it a versatile tool in our arsenal. This approach not only mirrors the nuances of human conversation but also significantly elevates NPC realism and engagement. By moving away from traditional, script-dependent game development, we pave the way for virtual characters that are not only more lifelike but also dynamically responsive to player interactions.

C. MEMORY RENEWAL MECHANISM

For LLMs, the upper limit of memory capacity is constrained by Tokens. The total Tokens consumed by the Prompt, plus the Tokens consumed by Completions, plus the Tokens consumed by Memory, must be less than a fixed Token limit, as constrained by the following:

$$T(P) + T(C) + T(M) \leq T_{limit} \tag{1}$$

We will mainly put the content related to formulas in the Scheduling Policy chapter. Through this mechanism, we allow the MemoryRepository to store more information without increasing the memory capacity.

1) MEMORY FORGETTING MECHANISM

The intricacies of human memory serve as a blueprint for our innovative Memory Renewal Mechanism within the MemoryRepository model. Human memory, despite the brain's limited capacity, demonstrates a remarkable ability to learn and retain vast amounts of information. This capability largely stems from the process of selective retention, where crucial knowledge is reinforced and less pertinent details are gradually forgotten. Long-term memory enhancement is a direct outcome of this selective forgetting and consolidation. In the realm of short-term memory, humans naturally prioritize significant information, allowing minor details to fade over time. This selective retention is crucial in preventing cognitive overload, ensuring the preservation of vital information, facilitating new skill acquisition, and discarding inconsequential or adverse experiences. Mimicking this process is essential for AI systems aiming to replicate human memory dynamics. It involves summarizing short-term memory and distinguishing the value of information to be retained or forgotten over time.

To emulate this selective forgetting in AI, we introduce a novel Scheduling Policy within our MemoryRepository. This strategy is designed to manage short-term memory by gradually forgetting information deemed less important. This mechanism not only simulates human memory processes but also enhances the AI's efficiency in handling and updating information, making it more relevant and context-aware in dynamic environments.

2) MEMORY ENHANCEMENT MECHANISM

The interplay between memory consolidation and memory decline is complex and synergistic. Summarization and forgetting in short-term memory do not function as two independent mechanisms but occur simultaneously. This simultaneous action is more consistent with the natural workings of human memory.

Specifically, short-term memory is subject to wear and tear through forgetting mechanisms, which purge irrelevant conversations to free up memory space for long-term memory. When this forgetting process occurs, the summarizing mechanism is also active. The formula that encapsulates this process is:

$$M = M_{meta} + M_l + M_s \tag{2}$$

M is the overall storage system, M_{meta} is initialization memory, which is used to control the fixed personality of AI NPC, M_s is short-term memory, M_l is long-term memory. We have a short-term memory room to store short-term memory, and a long-term memory room to store long-term memory, which are updated through our Scheduling Policy: at certain intervals, the contents of short-term memory will be summarized and then transferred to long-term memory. Then clear the contents of short-term memory to zero. This suggests that the summarizing and forgetting mechanisms are not sequential, but parallel processes, ensuring a more human-like memory system.

In the context of AI NPC memory, current information is stored in short-term memory chambers. This room can store up to ten rounds of dialogue. By the eleventh round, the previous ten conversations have gone through a process of summarizing and forgetting. The results of this process are then transferred to the long-term memory chamber, while the short-term memory that remains after forgetting remains in the short-term memory chamber.

IV. SCHEDULING POLICY OF MEMORYREPOSITORY

Based on the description in the previous section, in this section we focus on the algorithm and system modeling of MemoryRepository.

A. MEMORYREPOSITORY ALGORITHM

We propose the MemoryRepository algorithm. This is a set of mechanisms for AI-NPC to interact with MemoryRepository.The purpose is to allow AI-NPC to have a memory system capable of long-term interaction, making the interaction more Human-like.We provide an algorithmic description of the execution process for the MemoryRepository, as shown in Figure 2, and here is a description of the process:

Algorithm 1 Memory Repository Processing

```
Step1.Initialize a MemoryRoom for each NPC
  1.Create a short_term_memory_room to store all talks
  2.Create a long_term_memory_room to store all summaries
  3.Create a memory_room consisting of short_term_memory_room and long_term_memory_room
Step2.Memory Renewal and Memory interaction
while true do
  if MemoryInteraction detects a new talk occurs then
    Append talk to short_term_memory_room
  end if
  if length of short_term_memory_room is a multiple of $m$ then
    1.Extract the last m talks items from short_term_memory_room
    2.Summarize extracted items into one Summary
    3.Add Summary to long_term_memory_room
    4.Remove the extracted items from short_term_memory_room
  end if
  Through MemoryInteraction, the memory_room is output to the NPC according to the needs of the NPC.
end while
```

B. USER INTERACTION TASK MODELING

1) PLATFORM MODELING AND TASK MODELING

First, we introduce our system modeling, including platform model and task model. Our given experimental platform is represented as a tuple:

$$P = \langle LLM, T_{limit} \rangle \tag{3}$$

Among them, LLM and T_{limit} represent the LLMs used by the test platform and the maximum token of the LLMs interface adopted by the experimental platform.

For LLM, the maximum number of tokens is equivalent to the upper limit of LLM's memory. Considering the variable length of user conversation data and the fact that LLMs often impose token limits, we assume that is expressed as T_{limit}. Therefore, the number of conversations is limited by this token cap. The token limit is determined by the sum of prompt tokens, completions tokens, meta memory tokens, short-term memory tokens,long-term memory tokens, and meta memory tokens. Each aggregation session (represented by m) is subject to the following constraints:

$$T(P) + T(C) + T(M_l) + T(M_s) + T(M_{meta}) \leq T_{limit} \tag{4}$$

2) SINGLE DIALOGUE STATEMENT DESCRIPTION

For each interactive data t_i, we apply a forgetting function f designed to measure the importance of the data, allowing the AI to simulate human memory retention, which typically prioritizes important information. The forgetting function f is defined as follows:

$$f = e^{-\frac{t}{S}} \tag{5}$$

The variable t represents the number of rounds of interaction with the AI, while s quantifies the frequency of dynamic changes that occur between these rounds.

Second, for each unit of dialogue content, it is represented by t_i, which consists of two elements. We describe t_i as follows:

$$t_i = \left\langle f_i, T\left(t_i\right), Rl_{t_i} \right\rangle \tag{6}$$

where f_i, $T(t_i)$, and R_l represent the forgetting parameter of the current session, the token value of the current session and the importance parameter of the current statement,respectively.

3) SHORT-TERM MEMORY OPTIMIZATION METHODS

To mimic the natural tendency of human short-term memory to forget, we developed a specially designed method for selectively discarding the content of a user's short-term conversations. We define a summary set M_m for every m dialogue turns. For this set M_m we will use the function S to process the set and randomly forget several dialogues with minimum parameters f to the collection, thereby obtaining a new data set M' for long-term memory summary:

$$S\left(M_m, \min f_{t_i}\right) \rightarrow M' \tag{7}$$

4) LONG-TERM MEMORY OPTIMIZATION METHODS

We propose a long-term memory mechanism that simulates the human brain for long-term memory. People tend to remember memories that are repeated more deeply. Based on this principle, we will use LLM to summarize descriptions that repeat more than B and save them permanently.

For the content of memory, we divide the storage priority of conversations according to the following three levels:

$$R_l(b) = \begin{cases} ordinary & b = 1, \\ important & 1 < b < B, \\ forever & b \geq B, \end{cases} \tag{8}$$

In order to imitate the mechanism of human long-term memory, every once in a while(every m dialogue turns), we will summarize a *ordinary* conversation.Then the conversation priority of this summary will change to *important*. If the conversation continues to be repeated, we will consider this memory to be *forever*, and change the storage priority $R_l(t)$ of this conversation to *forever*. We employ Prompt Engineering to supply cues to LLMs, thereby enabling the summarization capability.

We model this operation:

$$M_l = \langle S(M_m), R_l \rangle \tag{9}$$

where M_l, M_m and R_l respectively represent the long-term memory, summary set M_m for every m dialogue turns, and the importance parameter of the current statement.Here is a simple implementation of a summary function prompt provided to LLM:*Please help me summarize the conversation between the NPC and the user. The format is as follows:""" NPC: How is the weather? User: The weather is sunny today. Summary of the conversation between the NPC and the user: The user knows that it is sunny today through the NPC.""" Now please help me summarize these conversations: M_m.*

C. SYSTEM-ORIENTED INDICATORS

In order to ensure that the AI's response comes from its memory, we introduce a parameter W to evaluate the completeness of the AI's answer after reflection. For a given set of n sets of session data $M = \{t_1, t_2, \ldots, t_n\}$, we use the following formula to define the data integrity percentage W: (Note: W value Higher indicates a more complete answer).

$$W(m) = \frac{\sum_0^n E(S(M_m), t_i)}{m} \tag{10}$$

where m represents the total number of conversations included in the conversation set M, each t_i corresponds to the content of the ith conversation, S represents the function to summary content of the conversation, E is the embedding function, used for evaluation Similarity between two conversations.

Of course, long conversations create a large memory burden, so the memory optimization of the evaluation method is an important metric. In addition to the above methods of evaluating statement completeness, we also introduce an additional metric, denoted E_m, to measure the memory optimization rate of a task. E_m is calculated as follows (note that the higher the E_m value, the better the memory optimization):

$$EM_M(m) = \frac{\sum_0^n T(t_i)}{T(sum^M)} \tag{11}$$

where M is the dialogue set used for evaluation, t_i is the i-th dialogue content in the data set, sum_m is the summary content of the set M, and T is the calculated dialogue token quantity function.

D. PROBLEM STATEMENT AND SCHEDULING OBJECTIVE

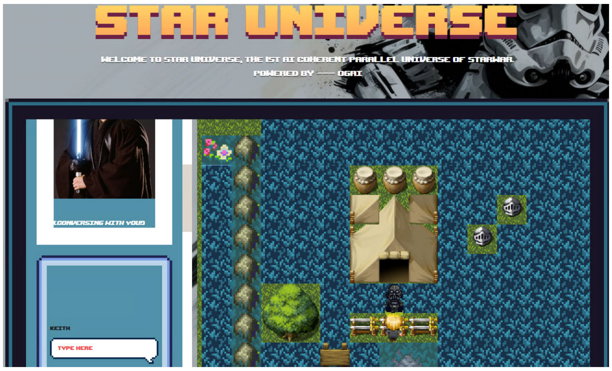
We provide a complete problem description here. The LLMs used in the given test platform P has token limitations T_{limit}, and the processing speed t_{token}^{LLM} of each token currently has a call data queue $\Gamma = \{t_1, t_2, \ldots\}$. For the number of summaries m of interactive data ti, we weigh the minimum processing time Rt and the LLMs tag limit T_{limit} to determine the optimal summarization strategy S and ensure the integrity of the data.

The scheduling problem in this paper can be formalized as follows

$$\begin{aligned}
&Given : P = \langle LLM, T_{limit} \rangle \quad and \quad \Gamma = \{t_1, t_2, \ldots\} \\
&find : S = \langle S_{1*}, S_{2*}, \ldots \rangle \\
&maximizing : W, Q \\
&subjected \quad to : s_{i*} = \left\langle m^i, f^i \right\rangle, i \in [1, 2, \ldots], \\
&\textstyle\sum_0^m T(t_i) \leq T_{limit} - T(P) - T(C) - T(S(M_m))
\end{aligned}$$

V. EXAMPLE: AN AI NPC GAME POWERED BY MEMORYREPOSITORY

As shown in Figure 3 and Figure 4, to visually demonstrate the effect of MemoryRepository in the field of AI NPC, we created an example called StarUniverse and case study through a series of controlled experiments, in which all NPCs are represented by LLM embedded in MemoryRepository.The example allows players to converse with AI NPC characters, each with their unique personality and design. Our analysis focuses on the ability of these AI characters to engage in long-term, human-like interactions.



STAR UNIVERSE

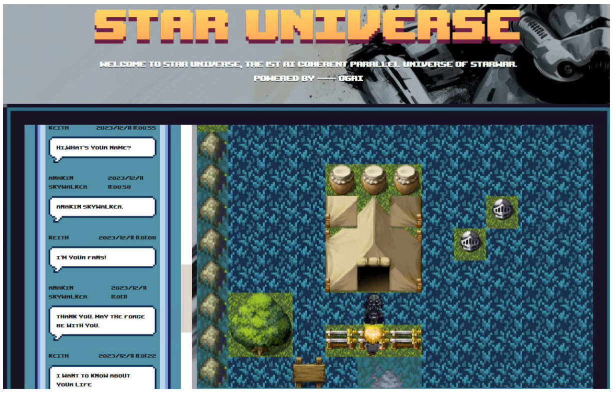
WELCOME TO STAR UNIVERSE, THE 1ST AI COHERENT PARALLEL UNIVERSE OF STARWAR.

POWERED BY —— OGAI

[CONVERSING WITH YOU]

KEITH

FIGURE 3. StarUniverse.



STAR UNIVERSE

WELCOME TO STAR UNIVERSE, THE 1ST AI COHERENT PARALLEL UNIVERSE OF STARWAR.

POWERED BY —— OGAI

KEITH 2023/12/11 11:00:55

HI,WHAT'S YOUR NAME?

ANAKIN SKYWALKER 2023/12/11 11:00:58

ANAKIN SKYWALKER.

KEITH 2023/12/11 11:01:08

I'M YOUR FANS!

ANAKIN SKYWALKER 2023/12/11 11:01:11

THANK YOU. MAY THE FORCE BE WITH YOU.

KEITH 2023/12/11 11:01:22

I WANT TO KNOW ABOUT YOUR LIFE

FIGURE 4. StarUniverse.

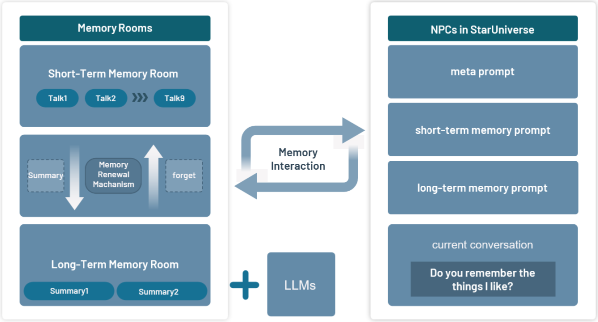


FIGURE 5. **Prompt Structure of NPC in MemoryRepository.**

We first describe the construction process of this case. The first step is to fine-tune the LLM using open-source training data related to NPCs dialogue. This fine-tuning process enhances the dialogue capabilities of NPCs, allowing them to provide users with more personalized and unique dialogue data. The second step is to integrate MemoryRepository into the example to achieve long-term interaction and human-like interaction functions. By incorporating a MemoryRepository into the example, NPCs can remember previous interactions and use that information to provide more personalized and context-aware responses.

After the case construction was completed, we used the case to conduct an intuitive demonstration, comparing the dialogue effects of NPCs without embedded MemoryRepository and those with embedded MemoryRepository.This demonstration will show whether MemoryRepository can enable LLM to enhance long-term conversations and make them more human-like interactions.

A. BUILDING OF EXAMPLE

1) FINE-TUNING

The first step in example development involved fine-tuning the LLM using open-source NPCs conversation data. This dataset comprehensively covers aspects such as the background, worldview, relationships, and personal experiences of NPCs. By incorporating these detailed layers, our approach equips AI NPC agents with the nuanced capability to accurately embody each character's unique personality, memories, and worldview. This enhanced representation leads to more authentic and engaging AI NPC dialogues.

Our fine-tuning method can be mathematically represented as follows:

$$W = W_0 + \Delta W . W \in R^{d\times k}, W_0 \in R^{d\times k} \tag{12}$$

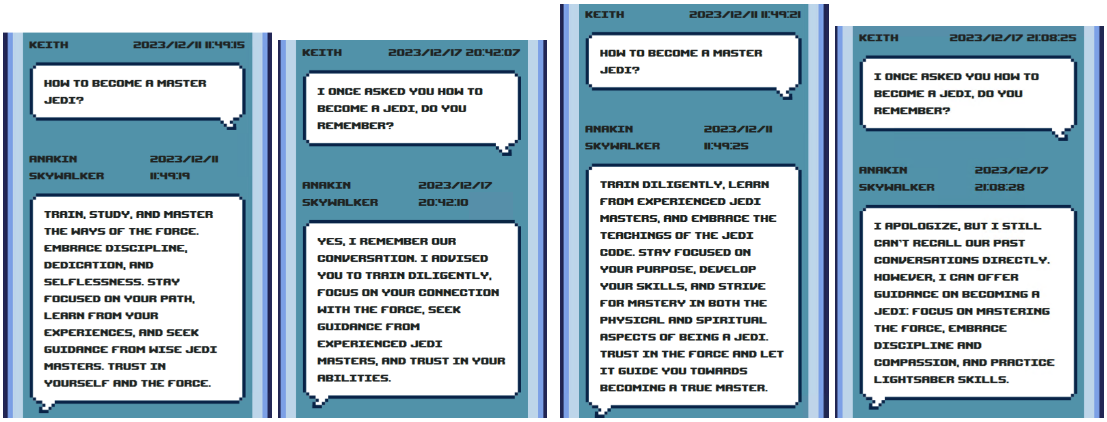
In this equation, W denotes the weight matrix after fine-tuning, and W_0 represents the original pre-trained weight matrix of the LLM before fine-tuning. The term ΔW signifies the adjustments made to the weights, encapsulating the gradient updates derived from the NPC conversation data. The dimensions $d \times k$ of these matrices specify the size of the weights, where d represents the number of features and k denotes the number of output dimensions in the model. These updates are critical as they modify the original model to more accurately reflect and respond to the complex characteristics and dialogues of NPCs in a gaming context.

2) POWERED BY MEMORYREPOSITORY

The second step entails incorporating the system into MemoryRepository. This integration endows AI NPC with dual memory storage capabilities within the MemoryRoom feature, encompassing both short-term and long-term memory.The Memory Renew Mechanism is a key component of our system, enabling NPCs to process, summarize, and selectively forget information, thereby lending a more authentic dimension to their interactions. These fundamental functionalities are the essence of MemoryRepository, providing AI NPCs with the capability for extended conversations and interactions that more closely resemble human behavior.

It is worth noting that the two-way interaction between the AI NPC agent and the memory system is implemented through LangChain in the form of Prompt Engineering. LangChain is a sophisticated tool or methodology designed to facilitate interactive communication processes. It allows AI NPCs to interact with their memory systems in a dynamic manner, significantly enhancing the realism of their responses and actions. In the domain of LLMs like ChatGPT, interaction is predominantly conducted through prompts and completions. A prompt is essentially the input given to an LLM, setting the context or query for its processing. The LLM responds to this input with what is known as completions, which are the outputs or answers generated by the model. For example, in gaming scenarios involving NPCs, prompts are carefully constructed to align with the NPCs' unique personalities and situational contexts. This interaction, facilitated by prompts and completions, is crucial in directing the LLM's responses, ensuring that the NPCs react in a manner that is consistent with their character profiles and the immediate narrative circumstances of the game environment. In reality, the method LLMs use to preserve memory involves a technique of prompt chaining. This means that when interacting with the LLM a second time, the input is not just the new prompt (p_2) but a concatenation of the first prompt (p_1), the response it generated (c_1), and then the second prompt (p_2). To illustrate this, suppose the initial input to the LLM, which is the first prompt, is p_1. This yields a response, denoted as c_1. For the subsequent interaction, when we input the second prompt p_2, the actual input the LLM receives is not merely p_2 but $p_1 + c_1 + p_2$. This approach effectively builds a continuous thread of interaction, allowing the LLM to 'remember' the earlier parts of the conversation and respond in a way that is coherent and contextually consistent with the ongoing narrative of the NPC.

Such a method ensures that NPCs in games not only react in accordance with their predefined personality traits and the immediate context but also demonstrate a continuity that mirrors human-like memory and conversational flow. As depicted in Figure 5, Here is the prompt structure for NPCs



KEITH 2023/12/11 11:49:15

HOW TO BECOME A MASTER JEDI?

ANAKIN SKYWALKER 2023/12/11 11:49:19

TRAIN, STUDY, AND MASTER THE WAYS OF THE FORCE. EMBRACE DISCIPLINE, DEDICATION, AND SELFLESSNESS. STAY FOCUSED ON YOUR PATH, LEARN FROM YOUR EXPERIENCES, AND SEEK GUIDANCE FROM WISE JEDI MASTERS. TRUST IN YOURSELF AND THE FORCE.

KEITH 2023/12/17 20:42:07

I ONCE ASKED YOU HOW TO BECOME A JEDI, DO YOU REMEMBER?

ANAKIN SKYWALKER 2023/12/17 20:42:10

YES, I REMEMBER OUR CONVERSATION. I ADVISED YOU TO TRAIN DILIGENTLY, FOCUS ON YOUR CONNECTION WITH THE FORCE, SEEK GUIDANCE FROM EXPERIENCED JEDI MASTERS, AND TRUST IN YOUR ABILITIES.

KEITH 2023/12/11 11:49:21

HOW TO BECOME A MASTER JEDI?

ANAKIN SKYWALKER 2023/12/11 11:49:25

TRAIN DILIGENTLY, LEARN FROM EXPERIENCED JEDI MASTERS, AND EMBRACE THE TEACHINGS OF THE JEDI CODE. STAY FOCUSED ON YOUR PURPOSE, DEVELOP YOUR SKILLS, AND STRIVE FOR MASTERY IN BOTH THE PHYSICAL AND SPIRITUAL ASPECTS OF BEING A JEDI. TRUST IN THE FORCE AND LET IT GUIDE YOU TOWARDS BECOMING A TRUE MASTER.

KEITH 2023/12/17 21:08:25

I ONCE ASKED YOU HOW TO BECOME A JEDI, DO YOU REMEMBER?

ANAKIN SKYWALKER 2023/12/17 21:08:28

I APOLOGIZE, BUT I STILL CAN'T RECALL OUR PAST CONVERSATIONS DIRECTLY. HOWEVER, I CAN OFFER GUIDANCE ON BECOMING A JEDI: FOCUS ON MASTERING THE FORCE, EMBRACE DISCIPLINE AND COMPASSION, AND PRACTICE LIGHTSABER SKILLS.

(a) StarUniverse Long-Term Interaction with MemoryRepository

(b) LLM Long-Term Interaction without MemoryRepository

FIGURE 6. Long-Term Comparable.

in StarUniverse can be formulated as:

$$P = P_m + P_s + P_l + P_t \tag{13}$$

P stands for Prompt. The P_m is a meta prompt, that corresponds to the character's personality, worldview, etc. P_s is a short-term memory prompt that will be passed into short-term memory for the LLMs. P_l is a long-term memory prompt, which will transfer long-term memory to LLMs, corresponding to the sum of the character's past short-term memory, which is long-term memory.Here is a P_t is the prompt representing the current conversation that will pass the current conversation to LLMs.

B. CASE STUDY

To demonstrate the Human-Like and Long-Term Interaction capabilities of the MemoryRepository, we integrated them into the conversation and compared them with Base-LLM without an embedded MemoryRepository. We have done a set of controlled experiments, one is an Example without embedded MemoryRepository, and the other is embedded. We will first demonstrate the intuitive effect. In the subsequent chapter, we will show a more scientific and comprehensive data result analysis to illustrate the advantages of the MemoryRepository.

1) LONG-TERM INTERACTIONS

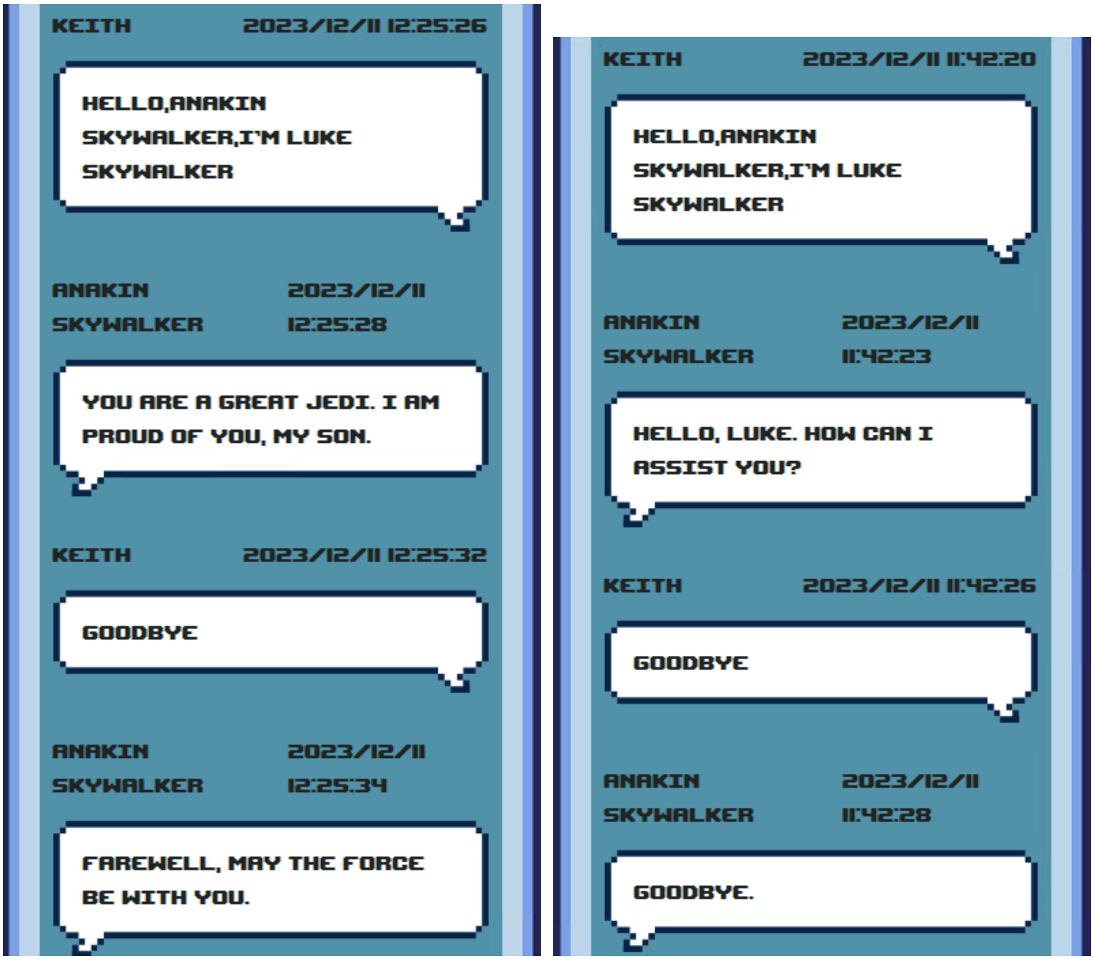
As shown in Figure 6, we communicated with the NPC in the game example and asked them the question, "How to become a Jedi Master?". At this point, regardless of whether a MemoryRepository was embedded or not, the responses from the NPCs were satisfactory and roughly similar. Then, we conducted a series of interrogations involving hundreds of unrelated records over several days. For display convenience, these dialogue records are hidden. Later, we asked again, "I once asked you how to become a Jedi Master, do you remember?". In cases where the LLM is embedded with a MemoryRepository, it can recall this previous question. However, in cases without a MemoryRepository, it is unable to recall the previous question. The results show that NPCs with an LLM embedded in a MemoryRepository can recall past interactions, but those without a MemoryRepository cannot. This demonstrates that an LLM embedded with a MemoryRepository is capable of performing long-term interactions compared to a base LLM.

2) HUMAN-LIKE

We used 400 pieces of dialogue data to test the Human-Like function of MemoryRepository. Here we show some representative pieces of dialogue data. As shown in Figure 7, Figure 7(a) uses LLM embedded with MemoryRepository for NPCs proxy, and Figure 7(b) uses base-LLM for NPCs proxy. We set the dialogue for the NPC named Anakin Skywalker as: "Hello, Anakin Skywalker, I'm Luke Skywalker." As humans, we know that both Anakin Skywalker and Luke Skywalker are Jedis, and Luke Skywalker is Anakin Skywalker's son. The NPC without a MemoryRepository responded mechanically, just like a standard LLM. In contrast, the NPC embedded with MemoryRepository responded with an awareness that Luke Skywalker is a Jedi and its son, even uttering phrases like "I am proud of you." Its farewell phrase was "May the Force be with you", a common saying among Jedis. These results vividly demonstrate that NPCs with an LLM embedded with a MemoryRepository can exhibit more human-like behaviors.

VI. EXPERIMENTS AND ANALYSIS

We evaluate our proposed MemoryRepository on a local multi-GPU server. Our primary objective is to assess the enhancement of performance in LLMs following the integration of MemoryRepository, with a specific emphasis on the improvement of capabilities in simulating NPCs for human-like and long-term interactions. We included a



KEITH 2023/12/11 12:25:26

HELLO,ANAKIN SKYWALKER,I'M LUKE SKYWALKER

ANAKIN SKYWALKER 2023/12/11 12:25:28

YOU ARE A GREAT JEDI. I AM PROUD OF YOU, MY SON.

KEITH 2023/12/11 12:25:32

GOODBYE

ANAKIN SKYWALKER 2023/12/11 12:25:34

FAREWELL, MAY THE FORCE BE WITH YOU.

(a) StarUniverse Human-Like with MemoryRepository

KEITH 2023/12/11 11:42:20

HELLO,ANAKIN SKYWALKER,I'M LUKE SKYWALKER

ANAKIN SKYWALKER 2023/12/11 11:42:23

HELLO, LUKE. HOW CAN I ASSIST YOU?

KEITH 2023/12/11 11:42:26

GOODBYE

ANAKIN SKYWALKER 2023/12/11 11:42:28

GOODBYE.

(b) LLM Human-Like without MemoryRepository

FIGURE 7. Human-Like Comparable.

series of tests on well-known LLMs such as ChatGLM, GPT-3.5, and GPT-4, to benchmark the performance of MemoryRepository on different platforms. Subsequently, we tested its robustness under various parameters.

In the following sections, we describe the configuration of the experimental environment, details of the experimental setup, and metrics to measure the success of the study. We then show the performance results of MemoryRepository when applied to various LLMs.

A. EXPERIMENTAL SETUP

Platform: We adopt the GPU platform as shown in Table 2.The experimental platform is built based on a server equipped with 20 CPU cores Intel(R) Xeon(R) E5-2640 CPU and 4 NVIDIA GeForce RTX 3090 GPUs. In addition, the server runs Ubuntu 20.04 LTS operating system and PyTorch 1.7.0 software environment.

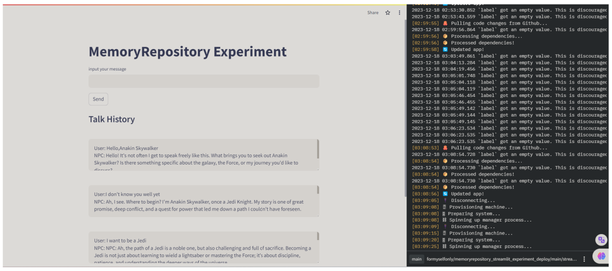
Dataset: We collected a series of open source NPC dialogue data from various fields on the Internet, and then organized them into a series of dialogue history records.

Validation set: We employed ChatGPT to formulate an initial set of 200 exploratory questions, each meticulously sourced from the Dataset. In parallel, we manually developed an additional set of 200 exploratory questions, ensuring they too were firmly grounded in the Dataset. The amalgamation of these 400 carefully curated questions constitutes the extensive verification set that underpins our research.

Evaluation Metrics: The performance of the model is evaluated based on the following metrics. (1) Memory retrieval accuracy: determines whether the relevant memory can be successfully retrieved (labels: 0: no;0.5:partial retrieved; 1: yes). (2) Response correctness: Evaluate whether the response contains the correct answer to the probe question (labels: 0: wrong; 0.5: partial correct; 1: correct). (3) Contextual coherence: Evaluate whether the structure of the response is natural and coherent, connecting the conversation context and retrieval memory (labels: 0: incoherent; 0.5: partially coherent; 1: coherent). (4) Human-like: The degree of human-like interaction manifested in the interactive dialogue process. (labels: 0: dissimilar of

TABLE 2. Platform Specifications.

	Parameters and Specifications	
Node	Rack Mount 2U J7VNFD2 Chassis	
System	Ubuntu 22.04, Pytorch 2.1.0	
CPU	Intel(R) Xeon(R) E5-2640 v4, 20C, 2.40GHz * 2/node	
Memory	32GB DDR4 Memory * 24 / node	
GPU	NVIDIA GeForce RTX 3090 GPU * 4	
LLM	GPT-4, GPT-3.5, ChatGPT-3	
LLM Parameters	TopP	1
	Temperature	0.7



MemoryRepository Experiment

input your message

Talk History

User: Hello,Anakin Skywalker
NPC: Hello! It's not often I get to speak freely like this. What brings you to seek out Anakin Skywalker? Is there something specific about the galaxy, the Force, or my journey you'd like to discuss?

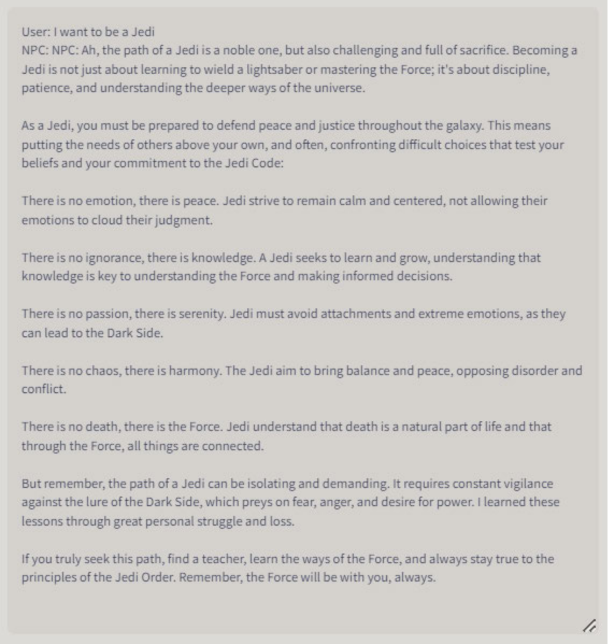
User:I don't know you well yet
NPC: Ah, I see. Where to begin? I'm Anakin Skywalker, once a Jedi Knight. My story is one of great promise, deep conflict, and a quest for power that led me down a path I couldn't have foreseen.

User: I want to be a Jedi
NPC: NPC: Ah, the path of a Jedi is a noble one, but also challenging and full of sacrifice. Becoming a Jedi is not just about learning to wield a lightsaber or mastering the Force; it's about discipline, patience, and understanding the deeper ways of the universe.

```
2023-12-18 02:53:30.852 `label` got an empty value. This is discouragec
2023-12-18 02:53:43.559 `label` got an empty value. This is discouragec
[02:59:55] Pulling code changes from Github...
2023-12-18 02:59:56.864 `label` got an empty value. This is discouragec
[02:59:56] Processing dependencies...
[02:59:56] Processed dependencies!
[02:59:58] Updated app!
2023-12-18 03:03:49.861 `label` got an empty value. This is discouragec
2023-12-18 03:04:13.284 `label` got an empty value. This is discouragec
2023-12-18 03:04:19.456 `label` got an empty value. This is discouragec
2023-12-18 03:05:01.748 `label` got an empty value. This is discouragec
2023-12-18 03:05:04.118 `label` got an empty value. This is discouragec
2023-12-18 03:05:04.119 `label` got an empty value. This is discouragec
2023-12-18 03:05:46.454 `label` got an empty value. This is discouragec
2023-12-18 03:05:46.455 `label` got an empty value. This is discouragec
2023-12-18 03:05:49.142 `label` got an empty value. This is discouragec
2023-12-18 03:05:49.144 `label` got an empty value. This is discouragec
2023-12-18 03:05:49.145 `label` got an empty value. This is discouragec
2023-12-18 03:06:23.534 `label` got an empty value. This is discouragec
2023-12-18 03:06:23.535 `label` got an empty value. This is discouragec
2023-12-18 03:06:23.535 `label` got an empty value. This is discouragec
[03:08:53] Pulling code changes from Github...
2023-12-18 03:08:54.728 `label` got an empty value. This is discouragec
[03:08:54] Processing dependencies...
2023-12-18 03:08:54.730 `label` got an empty value. This is discouragec
[03:08:54] Processed dependencies!
2023-12-18 03:08:54.730 `label` got an empty value. This is discouragec
[03:08:54] Processed dependencies!
[03:08:56] Updated app!
[03:09:05] Disconnecting...
[03:09:08] Provisioning machine...
[03:09:08] Preparing system...
[03:09:08] Spinning up manager process...
[03:09:09] Disconnecting...
[03:09:15] Provisioning machine...
[03:09:20] Preparing system...
[03:09:25] Spinning up manager process...
```

main
formyselfonly/memoryrepository_streamlit_experiment_deploy/main/strea...

FIGURE 8. StreamLit-Constructed Local Website for Experimental.



User: I want to be a Jedi
NPC: NPC: Ah, the path of a Jedi is a noble one, but also challenging and full of sacrifice. Becoming a Jedi is not just about learning to wield a lightsaber or mastering the Force; it's about discipline, patience, and understanding the deeper ways of the universe.

As a Jedi, you must be prepared to defend peace and justice throughout the galaxy. This means putting the needs of others above your own, and often, confronting difficult choices that test your beliefs and your commitment to the Jedi Code:

There is no emotion, there is peace. Jedi strive to remain calm and centered, not allowing their emotions to cloud their judgment.

There is no ignorance, there is knowledge. A Jedi seeks to learn and grow, understanding that knowledge is key to understanding the Force and making informed decisions.

There is no passion, there is serenity. Jedi must avoid attachments and extreme emotions, as they can lead to the Dark Side.

There is no chaos, there is harmony. The Jedi aim to bring balance and peace, opposing disorder and conflict.

There is no death, there is the Force. Jedi understand that death is a natural part of life and that through the Force, all things are connected.

But remember, the path of a Jedi can be isolating and demanding. It requires constant vigilance against the lure of the Dark Side, which preys on fear, anger, and desire for power. I learned these lessons through great personal struggle and loss.

If you truly seek this path, find a teacher, learn the ways of the Force, and always stay true to the principles of the Jedi Order. Remember, the Force will be with you, always.

FIGURE 9. A segment of these historical records.

human; 0.5: partially similar of human; 1: completely similar of human)).

B. EVALUATION

The primary focus centers on assessing the capabilities of these models in simulating human-like and extended interactions within the role of AI NPC agents, particularly highlighting the enhancements brought about by their integration with MemoryRepository.

Our experimental approach is as follows: In the first step, we prompt the LLMs to respond to the exploratory questions in the validation set, thereby generating a series of answer history records.As shown in Figure 8,We constructed a local website using StreamLit to conduct tests and display results of experimental question-and-answer sessions.A segment of these historical records is depicted in Figure 9. In the second step, our focus shifted to the evaluation of these previously generated response histories. We enlisted human annotators to critically assess the model's responses to these exploratory queries, scoring the model based on established evaluation metrics. The mean of these scores was then calculated to serve as the outcome. Following this, multiple experimental iterations were conducted, with the scores obtained therein being selected as the final score. To ensure the accuracy of the data, we conducted multiple rounds of experiments and expanded the scope of our experimentation by testing and evaluating adjustments to the parameters of large prediction models. This expansion is crucial for a deeper and more comprehensive understanding of MemoryRepository's performance across different models. It also enables us to assess its adaptability and scalability in various environments.

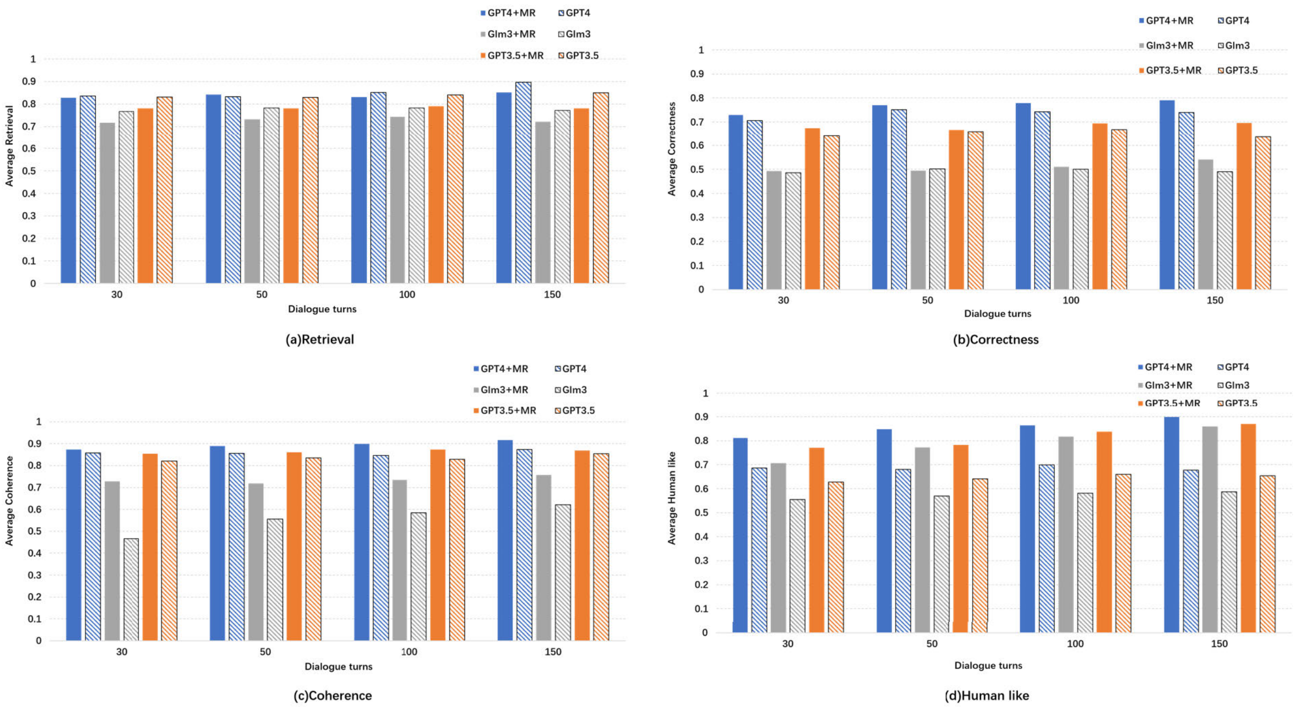
1) MEMORYREPOSITORY PERFORMANCE AND COMPARISON

In this section, we endeavor to critically analyze the performance disparities exhibited by LLMs when operational as AI agents for NPCs. For model selection, we strategically chose to evaluate MemoryRepository using three well-known LLMs: GPT-4, GPT-3.5, and ChatGLM. ChatGLM distinguishes itself as a lightweight model, optimized for conversational AI, and boasting robust multilingual support.Following this, we explore GPT-3.5, the forerunner of GPT-4, developed by OpenAI. It is celebrated for its broad spectrum of language understanding and generation, pivotal in orchestrating complex, text-based interactions. Lastly, we integrate OpenAI's advanced model, GPT-4, into our evaluation. GPT-4 outshines its antecedents with superior multilingual processing abilities and intricate problem-solving capacities, exhibiting exceptional versatility for a range of linguistic applications.

The results of Figure 10(a) and Figure 10(b) show that the integration of the MemoryRepository (MR) into the system, when compared with the baseline model, did not yield a significant enhancement in the correctness indicator, and it was observed that the retrieval indicator experienced a decline.This performance degradation may be due to the introduction of a forgetting mechanism into the algorithm of this article, which selectively eliminates information that is considered less important, and only retains the summary memory and a part of the more repeated memory dialogue, so the memory is correct but not very detailed. However, for AI NPC, what we need more is Long-Term Interaction and Human-Like Interaction, which are determined by the two indicators of Correctness and Human-Like.As shown in Figure 10(c), the ChatGPT baseline model is significantly better

TABLE 3. Comparison of MemGPT and MemoryRepository-embedded GPT-3.5 Performance in the 100th Round of Dialogue.

LLM	Retrieval	Correctness	Coherence	Human Like
$MemGPT$	0.831	0.713	0.874	0.718
$MemoryRepository - embeddedGPT3.5$	0.806	0.705	0.886	0.861



(a)Retrieval

(b)Correctness

(c)Coherence

(d)Human like

FIGURE 10. Comparative analysis of MemoryRepository strategy and original model performance.

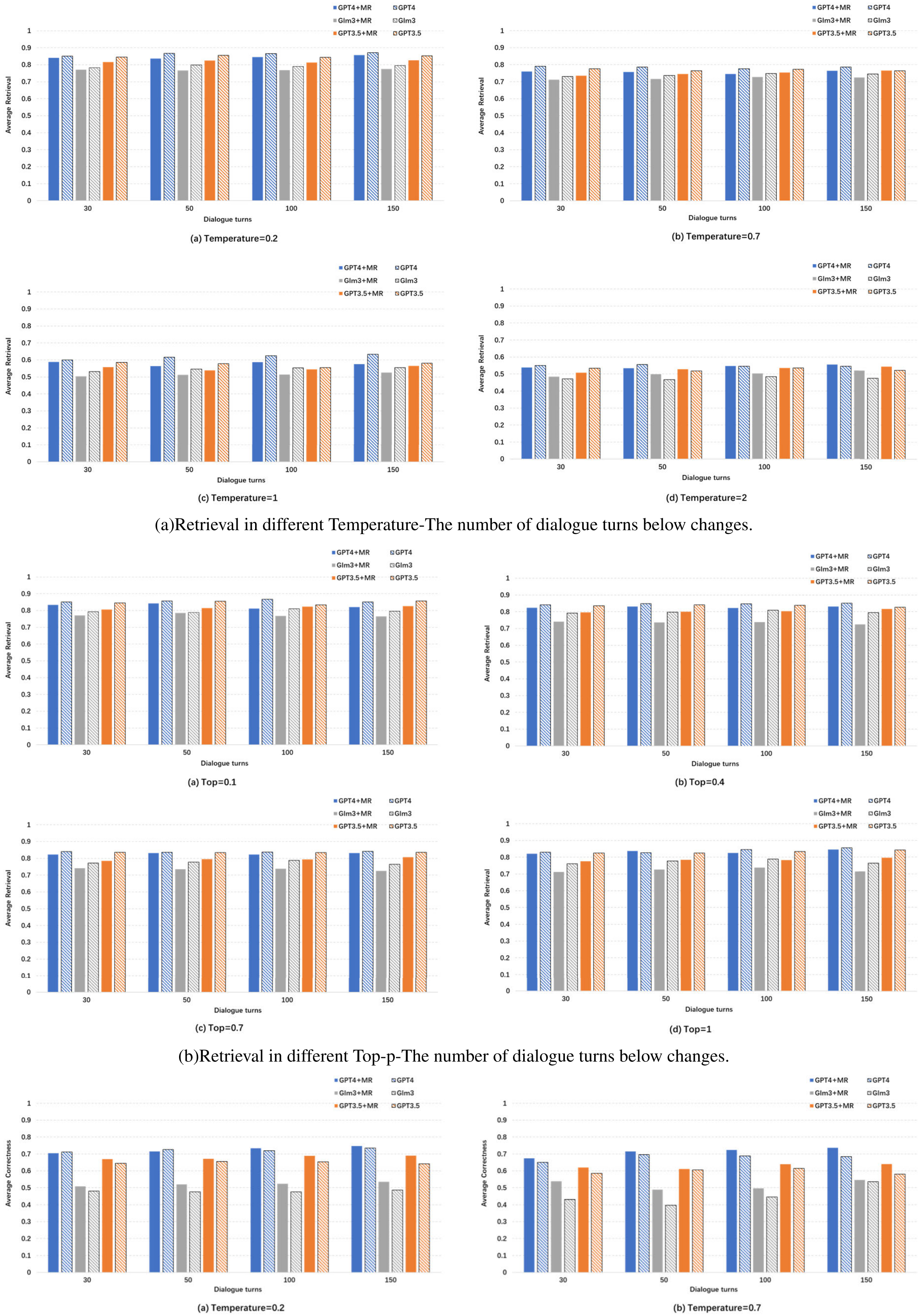
than ChatGLM in this indicator, and MP can significantly improve chatGLM. This is because the premade prompt summary we provide can Improves conversational coherence because it provides clear context for communication, guides topic direction, limits the scope of ambiguity, and enables language models to more effectively predict and generate relevant responses. At the same time, it maintains topic consistency and prevents conversations from straying from the main line. In Figure 10(d), we can see that MemoryRepository has made substantial progress in simulating human interaction compared to the basic model. Over time, the responses generated using the MemoryRepository strategy gradually became more similar to real human interactions, demonstrating the effectiveness of MemoryRepository in optimizing the quality of interactions. At the same time, due to the lack of similar optimization mechanisms, the performance of the baseline model in terms of similarity does not change much over time. Furthermore, we clearly observe that GPT-4 outperforms GPT-3.5, which in turn outperforms ChatGPT-3.

2) PERFORMANCE COMPARISON OF MEMORYREPOSITORY AND MEMGPT

To further assess performance, we compared the performance of MemGPT and GPT-3.5 embedded with Memory Repository in the 100th round of dialogue,as shown in Table 3, the results indicate that MemGPT performs similarly to MemoryRepository in terms of correctness and coherence, with its Retrieval is slightly better than MemoryRepository. However, its Human-Like score is notably lower than MemoryRepository.This is because MemGPT does not incorporate the forgetting and summarizing mechanisms of human memory, resulting in no performance decline in Retrieval and no improvement in Human-Like performance. This shows that in games where NPC's Human-Like is the main indicator, MemoryRepository has better overall performance.

3) PERFORMANCE COMPARISON OF MEMORYREPOSITORY-ENHANCED LLMS ACROSS DIVERSE INFERENCE PARAMETERS

In this section, we aim to evaluate the robustness of LLMs on two performance metrics: correctness and retrieval. To this end, we conducted experiments by adjusting the two main parameters of the model - temperature and top-p parameters. When testing the impact of temperature parameters on performance, we fixed the top-p parameter at 1 and selected four different temperature parameter settings: 0.2, 0.7, 1, and 2. These settings are designed to explore differences



(a) Temperature=0.2

(b) Temperature=0.7

(c) Temperature=1

(d) Temperature=2

(a)Retrieval in different Temperature-The number of dialogue turns below changes.

(a) Top=0.1

(b) Top=0.4

(c) Top=0.7

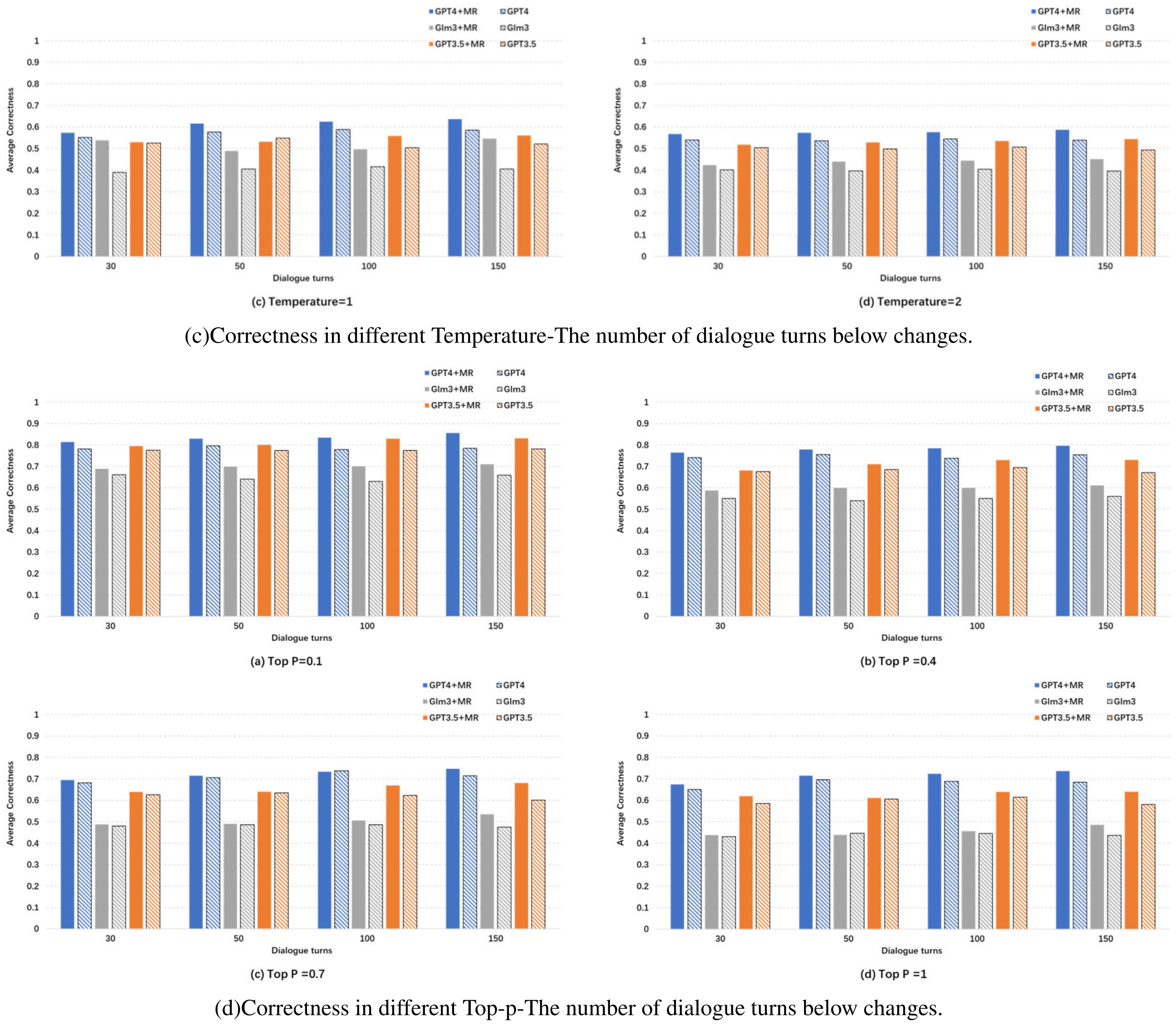
(d) Top=1

(b)Retrieval in different Top-p-The number of dialogue turns below changes.

(a) Temperature=0.2

(b) Temperature=0.7

FIGURE 11. Performance Comparison of MemoryRepository-Enhanced LLMs Across Diverse Inference Parameters.



(c) Temperature=1

(d) Temperature=2

(c)Correctness in different Temperature-The number of dialogue turns below changes.

(a) Top P=0.1

(b) Top P =0.4

(c) Top P =0.7

(d) Top P =1

(d)Correctness in different Top-p-The number of dialogue turns below changes.

FIGURE 11. ***(Continued.)*** **Performance Comparison of MemoryRepository-Enhanced LLMs Across Diverse Inference Parameters.**

in model performance at different sampling temperatures. In contrast, when evaluating the impact of the top-p parameter on performance, we fixed the temperature parameter at 0.7 and selected four different top-p parameter settings: 0.1, 0.4, 0.7, and 1. This is to analyze how the model's output diversity affects its performance under different probability cutoff points. The following sections detail the experimental design, analysis of the results, and the conclusions we draw from them.

Figure 11(a) illustrates that with the escalation of the temperature parameter, the information retrieval (Retrieval) performance of LLMs exhibits a significant decline. Notably, this decline is seemingly invariant to the number of dialogue turns. Additionally, the integration of the Memory Repository marginally affects the retrieval performance across different LLMs. Nonetheless, at a temperature setting of 2, the influence of MemoryRepository on retrieval performance is virtually imperceptible. Figure 11(b) demonstrates that the information retrieval performance of LLMs remains constant with increasing top-p parameter values, suggesting a negligible effect of the top-p parameter on retrieval efficacy. Figure 11(c) presents a notable decrease in the correctness metric for LLMs as the temperature parameter rises. Despite this, the models display a progressive increase in correctness with the accumulation of dialogue turns, even under elevated temperature conditions. Figure 11(d) indicates a slight decrement in correctness as the top-p parameter is augmented. Concurrently, there is an observed incremental improvement in correctness with an increased number of dialogue rounds.

The comparative analysis of model performance includes preliminary indications that GPT-4 ranks highest in terms of accuracy, followed by GPT-3.5, with GLM3 in third place. Additionally, the performance of MR is almost unaffected by different hyperparameters, suggesting that MR possesses sufficient stability.

VII. CONCLUSION

We present MemoryRepository, a dedicated memory mechanism designed for LLMs to enhance their adaptability to AI NPC. MemoryRepository enables LLM agents to mimic human-like capabilities of summarizing and forgetting information. To visually demonstrate its effectiveness, we utilize a Example where all NPCs are represented by AI, showcasing the impact of MemoryRepository. By incorporating MemoryRepository and fine-tuning the NPCs dialogue data, LLM agents exhibit remarkable long-term dialogue interactions while maintaining a more human-like conversational style. With the implementation of these two fundamental functions, LLM agents embedded with MemoryRepository can engage in extended interactions without being constrained by memory limitations. Furthermore, it significantly enhances the human-like qualities of AI NPC agents. We conducted an experiment using 400 test data samples to evaluate MemoryRepository's advancements in human-Like capabilities and Long-Term Interaction capabilities. MemoryRepository's versatility is exemplified by its compatibility with both open-source models like ChatGLM and closed-source models like ChatGPT 3.5 and ChatGPT 4.

In conclusion, MemoryRepository proves to be a valuable addition to Language Model agents, enabling them to mimic human-like abilities to summarize and forget information. By incorporating MemoryRepository, LLM agents can engage in extended interactions and maintain a more natural conversational style. Its compatibility with various models further enhances its versatility and applicability in different scenarios.

REFERENCES

[1] M. O. Riedl and V. Bulitko, "Interactive narrative: An intelligent systems approach," *AI Mag.*, vol. 34, no. 1, pp. 67–77, Mar. 2013.

[2] R. Koster, *Postmortems From Game Developer: Insights From the Developers of Unreal Tournament, Black and White, Age of Empire, and Other Top-Selling Games*. Boca Raton, FL, USA: CRC Press, 2019.

[3] R. Prada and A. Paiva, "Teaming up humans with autonomous synthetic characters," *Artif. Intell.*, vol. 277, pp. 103–167, Jan. 2019.

[4] R. P. Pérez, "Creativity in intelligent technologies and the non-player characters' dilemma," in *Proc. 10th Int. Conf. Comput. Creativity*, 2019, pp. 3–10.

[5] M. Treanor, B. Blackford, M. Mateas, and I. Bogost, "Game-O-matic: Generating videogames that represent ideas," in *Proc. 3rd Workshop Procedural Content Gener. Games*, 2012, pp. 1–8.

[6] A. Modarressi, A. Imani, M. Fayyaz, and H. Schütze, "RET-LLM: Towards a general read-write memory for large language models," 2023, *arXiv:2305.14322*.

[7] J. S. Park, J. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein, "Generative agents: Interactive simulacra of human behavior," in *Proc. 36th Annu. ACM Symp. User Interface Softw. Technol.*, 2023, pp. 1–22.

[8] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, and Z. Dong, "A survey of large language models," 2023, *arXiv:2303.18223*.

[9] C. Zhang, C. Zhang, S. Zheng, Y. Qiao, C. Li, M. Zhang, S. K. Dam, C. M. Thwal, Y. L. Tun, L. L. Huy, D. Kim, S.-H. Bae, L.-H. Lee, Y. Yang, H. T. Shen, I. So Kweon, and C. S. Hong, "A complete survey on generative AI (AIGC): Is ChatGPT from GPT-4 to GPT-5 all you need?" 2023, *arXiv:2303.11717*.

[10] R. Gallotta, G. Todd, M. Zammit, S. Earle, A. Liapis, J. Togelius, and G. N. Yannakakis, "Large language models and games: A survey and roadmap," 2024, *arXiv:2402.18659*.

[11] W. Zhou, X. Peng, and M. Riedl, "Dialogue shaping: Empowering agents through NPC interaction," 2023, *arXiv:2307.15833*.

[12] Y. Liu, T. Han, S. Ma, J. Zhang, Y. Yang, J. Tian, H. He, A. Li, M. He, Z. Liu, Z. Wu, L. Zhao, D. Zhu, X. Li, N. Qiang, D. Shen, T. Liu, and B. Ge, "Summary of ChatGPT-related research and perspective towards the future of large language models," *Meta-Radiol.*, vol. 1, no. 2, Sep. 2023, Art. no. 100017.

[13] W. Wang, L. Dong, H. Cheng, X. Liu, X. Yan, J. Gao, and F. Wei, "Augmenting language models with long-term memory," 2023, *arXiv:2306.07174*.

[14] H. Panwar, "The NPC AI of the last of us: A case study," 2022, *arXiv:2207.00682*.

[15] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[16] E. Parisotto and R. Salakhutdinov, "Neural map: Structured memory for deep reinforcement learning," 2017, *arXiv:1702.08360*.

[17] J. Weston, E. Dinan, and A. H. Miller, "Retrieve and refine: Improved sequence generation models for dialogue," 2018, *arXiv:1808.04776*.

[18] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, and J. Oh, "Grandmaster level in StarCraft II using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, Nov. 2019.

[19] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and A. Askell, "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1877–1901.

[20] J. Achiam et al., "GPT-4 Technical Report," 2024, *arXiv:2303.08774*.

[21] S. Sukhbaatar, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.

[22] A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwinska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou, A. P. Badia, K. M. Hermann, Y. Zwols, G. Ostrovski, A. Cain, H. King, C. Summerfield, P. Blunsom, K. Kavukcuoglu, and D. Hassabis, "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, pp. 471–476, Oct. 2016.

[23] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap, "Meta-learning with memory-augmented neural networks," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2016, pp. 1842–1850.

[24] J. Rae, J. J. Hunt, I. Danihelka, T. Harley, A. W. Senior, G. Wayne, A. Graves, and T. Lillicrap, "Scaling memory-augmented neural networks with sparse reads and writes," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[25] C. Packer, S. Wooders, K. Lin, V. Fang, S. G. Patil, I. Stoica, and J. E. Gonzalez, "MemGPT: Towards LLMs as operating systems," 2023, *arXiv:2310.08560*.

[26] A. Adhikari, A. Ram, R. Tang, and J. Lin, "Rethinking complex neural network architectures for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2019, pp. 4046–4051.

[27] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," 2019, *arXiv:1901.02860*.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol. (NAACL-HLT)*, vol. 1, 2019, p. 2.

[29] D. A. Hudson and C. D. Manning, "GQA: A new dataset for real-world visual reasoning and compositional question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6693–6702.

SHIJIE ZHENG (Student Member, IEEE) is currently pursuing the master's degree in artificial intelligence with Yangtze University. His research interests include AI-generated content (AIGC), with a specific interest in the application of large language models (LLMs) in the domain of non-player characters (NPCs) in gaming. His work aims to enhance interactive gaming experiences through advanced AI techniques.

KEITH HE received the B.A. degree in e-commerce from Guangdong University of Foreign Studies, Guangzhou, China, in 2006. He has been associated with corporate industry for more than 15 years. He has operated in various senior roles in the field of public cloud and ICT. He is currently the Product Director in a world's leading cloud organization. He has a vast experience implementing various technical projects using advanced risc machine (ARM) infrastructure. His research interests include cloud gaming platform and AI-generated content (AIGC).

LE YANG received the B.S. degree in network engineering from Guangdong University of Technology, Guangzhou, China, in 2021, where he is currently pursuing the master's degree in computer technology. His research interest includes parallel computing.

JIE XIONG received the B.S. degree in computer communication from Chongqing University of Posts and Telecommunications, China, in 1998, the M.S. degree in computer science from Yangtze University, China, in 2005, and the Ph.D. degree in geophysics and information technology from China University of Geosciences, China, in 2012. He is currently a Professor with the School of Electronics and Information, Yangtze University. His research interests include computer application, applied geophysics, and artificial intelligence.