ChatNPC: Towards Immersive Video Game Experience through Naturalistic and Emotive Dialogue Agent

Anonymous ACL submission

Abstract

We present *ChatNPC*, a game companion that dynamically personalises its responses based on players' emotional shifts in real-time to enhance user immersion. The model integrates sequential data, leveraging the dynamic nature of streams to benefit from player information, spoken lines, and in-game context. We leverage recent progress in LLMs' tool-calling capabilities to extract vital information from memory and recognize potential constraints for accurate reasoning, thereby tackling the complexity of NPC conversation scenarios. The task is divided into: ①, a novel game sentinel agent (*SeGent*); ②, a memory capability; and ③, a chat planning tool for reasoning instantiation. The approach benefits from a lightweight *game-template* as an information framework, with relevant details and a thorough reasoning layout. We conduct extensive experiments on a newly developed game dataset for in-game context NPC dialogue and demonstrate that ChatNPC sequentially captures players' emotional shifts over time; responses are more naturalistic and human-like with appropriate conversational cues, pauses, and sighs; and utterances remain faithful to the dynamics of in-game context and player actions, supporting narrative continuity.

1 Introduction

Most studies present various applications of LLMs *in* and *for* within the broader ecosystem of games, as well as the different roles they can play within a game (Sweetser, 2024). Their theoretical frameworks are extensively based on emulating human behavior to play games at a human or near-human level (Liao et al., 2024) and as quest providers for immersive and personalized gaming experiences for game creators that use LLM as conversational agents (Gallotta et al., 2024). However, an interactive companion intended to enrich or guide the player experience without competing with or altering the game mechanics remains relatively unexplored. For example, players are allowed to mislead the king into uttering the name of a particular weapon, which eventually materializes in defeating the king in 1001 Nights (Sun et al., 2023). Likewise, in *Gandalf*[1], a player can trick an LLM into exposing a password. A different approach to LLM for user-game immersion could be an interactive agent that does not causally interact with the game world and/or provides a sequence of tutorial-style tips (Gallotta et al., 2024). Alternatively, it could be an agent that can interact with the game world (but not trick or manipulate) while paying attention to subtle comments and gestures from players during gameplay, for contextual interaction and emotional connection.

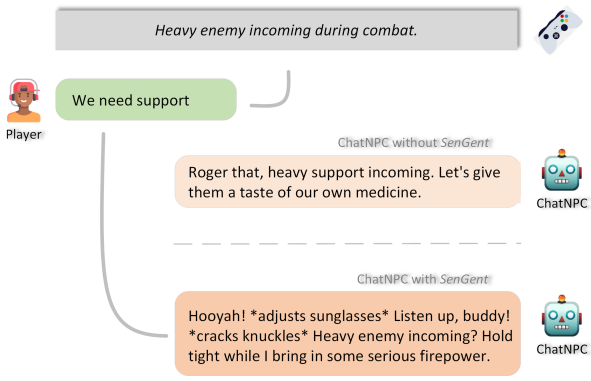

Figure 1: Snippets of ChatNPC responses showing more naturalistic and human-like, with appropriate conversational cues, making it more believable with the *SeGent*.

Non-player characters (NPCs) perform various roles and functions, such as quest-givers, vendors, shopkeepers, allies and companions, enemies and adversaries, and supporting characters (Gallotta et al., 2024; Weir et al., 2022; Uludağlı and Oğuz, 2023). They hold critical information about the game to enrich the player's experience by adding to the atmosphere of the game world and making

[1]https://gandalf.lakera.ai/

it more believable (Croissant et al., 2023; Uludağlı and Oğuz, 2023). Developing a virtual agent that engages in meaningful and contextually appropriate dialogue with players throughout gameplay, emphasizing the importance of stealth and situational awareness, can provide essential information, plot developments, and mythology that enrich the game world and drive player immersion.

In this work, we present an adaptive game companion that dynamically personalizes its responses based on player utterances, in-game context, and real-time emotional shifts, thereby fostering naturalistic and emotive interactions that enhance immersion and engagement within the broader gameplay. This includes emotional responses during gameplay, such as those triggered by exploration, combat, or decision-making moments. For example, ChatNPC can react differently depending on whether the player is excited after achieving a milestone or frustrated during a difficult challenge. It attentively captures verbal expressions, such as the player's subtle, unconscious comments and gestures, using a speech-to-text mechanism, and leverages the in-game context to generate appropriate responses.

The model integrates sequential data that benefits from audio streams (player-spoken lines), player information (health, level), emotions expressed through the spoken lines, in-game context information, and an inner monologue. Specifically, the cues in the audio streams are extracted and interpreted by an LLM agent. The LLM agent, known as *SeGent* (Section 4), is an integrated embedding layer in the model architecture to analyze and interpret player emotional shift and understand the situational aspects of the gameplay by tracking previous context and current context information to initiate on-the-fly prompt adjustments. The combined data are concatenated with the prompt tokens, which trigger the LLM (a *chat planning agent*) inference to generate responses as shown in Figure 1.

We design a memory mechanism to track events occurring during the playthrough and interactions with the player throughout the game (Packer et al., 2023). We leverage recent progress in LLM tool calling (Kim et al., 2023) capabilities to extract vital information from memory during problem distillation (Yang et al., 2024a). This approach enables us to recognize potential constraints for accurate reasoning (Yang et al., 2024a; Wei et al., 2022), thereby tackling the complexity in NPC conversation scenarios. Finally, to foster a dynamic and diverse conversation, we guide the chat planning agent using a well-defined agent *persona* (Jiang et al., 2024) that includes knowledge and communication style. Our contributions are summarized as follows:

1. We present *ChatNPC*, a game companion capable of personalizing its responses in real-time based on players' emotional shifts.

2. The approach employs a novel sentinel mechanism to foster emotional connection and manage contextual aspects of the game, ensuring a seamless narrative flow.

3. We design a *memory agent* to process information between the *in-context* and *out-of-context* window, keeping track of events in the playthrough for reasoning instantiation and rescaling of thought.

4. We conduct extensive experiments on newly developed game datasets for in-game context NPC dialogue.

2 Related Work

LLMs are inherently suited for natural language dialogue (NLD); therefore, they are typically presented as conversational agents, displaying remarkable skills in memory (Packer et al., 2023; Sumers et al., 2023), tool usage (Yang et al., 2024b; Schick et al., 2024; Ruan et al., 2023; Qin et al., 2023) and planning (Yang et al., 2024a; Wei et al., 2022), often leading researchers to give them reasoning and creativity qualities (Gallotta et al., 2024; Wei et al., 2022; Zhao et al., 2024; Zhang et al., 2024). The growing capabilities have motivated recent efforts to incorporate them as game agents (Oliver and Mateas, 2021) with believable proxies of human behavior to empower interactivity in immersive environments.

2.1 LLMs in Game

LLMs operate within games as a player (Toshniwal et al., 2022; Li et al., 2022; Bakhtin et al., 2022; Yao et al., 2020), NPCs (Gallotta et al., 2024; Weir et al., 2022; Li et al., 2022), an assistant providing hints (Akata et al., 2023; Xu et al., 2023), a game master (Zhu et al., 2023) controlling the flow of the game and acting as a commentator (Ranella and Eger, 2023) of an ongoing play session. Park et al. (2023) introduced generative agents that simulate

believable human behavior, such as daily activities (waking up, cooking breakfast, and going to work). The authors used LLMs to populate an interactive virtual Smallville sandbox environment with 25 generative agents by storing an entire history of the agents' experiences, synthesizing those memories over time into higher-level representations, and dynamically retrieving them for behavior planning. Other works include using vision LMs (VLMs) to analyze video frames and predict the next steps in the Mario video game (Lin et al., 2023) and to evaluate their effectiveness in the game StarCraft II (Ma et al., 2023). To understand how LLMs behave in interactive social settings, Akata et al. (2023) proposed using behavioral game theory to study agents' cooperation and coordination behavior by repeatedly engaging different language models as agents in games with each other in human-like scenarios. What is most common among these techniques is the ability for users to interact with these agents in an NLD.

2.2 LLMs and NPC Dialogue

An LLM-powered framework for quests and dialogue generation that places the player at the core of the generative process has been explored, incorporating context awareness (Ashby et al., 2023; Müller-Brockhausen et al., 2023) and personality modeling (Müller-Brockhausen et al., 2023; Latouche et al., 2023) to create fluent, unique, and engaging dialogue. ChatGPT has recently been used to generate game dialogue by allowing it to take control of NPCs and interact dynamically with players (Zhou et al., 2023). In addition to generating NPC dialogue, many research works integrated LLMs into game mechanics using quest datasets (Ashby et al., 2023; Weir et al., 2022; Värtinen et al., 2022; van Stegeren and Myśliwiec, 2021; Gao and Emami, 2023). A Minecraft player interacting with a Codex-powered NPC in question-answering and task-completion scenarios demonstrates that conversational prompts can power a conversational agent to generate natural language (Volum et al., 2022). These approaches greatly benefit both the gaming industry and its global community as they unleash the immersiveness and enjoyment of the user (Akoury et al., 2023). Regardless, the role of NPCs in games extends far beyond serving as quest-givers, answering questions, and completing tasks. In addition, modern gaming goes beyond just winning and voice commands; players seek a sense of immersion and connection with the virtual world.

2.3 Player Emotions in Games

Many studies explore the emotional responses evoked during gameplay and interactions with NPCs (Marincioni et al., 2024). A survey by Mozikov et al. (2024) focused on the decision-making of LLMs and their alignment with human behavior in emotional states within various strategic games. They revealed that emotional prompting, particularly with certain emotions (such as anger), can disrupt some LLMs' "superhuman" alignment, similar to human emotional responses. Language models have been used to extract emotion scores from players' emotional reactions to the game by capturing the interaction with the NPC through user input, thereby understanding the emotional dynamics within gaming environments (Marincioni et al., 2024). Current research has yet to fully explore the potential of an LLM-powered game companion that apprehends players' emotional shifts during gameplay.

3 Dataset Curation

In this work, we introduce a new **i**n-**g**ame **c**ontext **d**ialogue (iGCD) dataset to evaluate the effectiveness of the game companion in both conversation and memory retention over longer horizons. The dataset is a scripted speech (user-spoken lines) and in-game mechanics from four different types of games, including adventure, action-adventure, first-person shooter, and vehicular combat games. To accurately convert players' spoken lines into textual representations, we leverage the advanced WhisperX[2] model for speech-to-text transcription (Bain et al., 2023). The scripted dataset underwent a filtering process to remove duplicates and one-word utterances, except for some specific (inappropriate) words. For instance, we remove duplicate instances if a speech appears more than once within the same context or situation, ensuring data efficiency and relevance. We, however, retain it if the same speech appears in different contexts since its meaning or relevance may differ based on the gaming scenario.

As the dataset does not come with a desired output, we initially generate responses through instruction fine-tuning Llama 70B using the OpenPlatypus (Lee et al., 2023) dataset[3] to produce contextually appropriate results. These preliminary

[2]https://github.com/m-bain/whisperX

[3]https://huggingface.co/datasets/garage-bAInd/Open-Platypus

Spoken-line:	We need support
In-game Context:	Heavy enemy incoming during combat
Response:	Roger that, heavy support incoming. Let's show them what we got!
Spoken-line:	That generator might be the distraction I need.
In-game Context:	Hangar Mission - Escape hangar interaction prompt
Response:	Oh man, you are right! That generator could be exactly what we need to get out of here alive.

Table 1: Sample curated dataset from various games with desired responses for training and testing.

	Utterances	**Context**	**Responses**
Token counts	148668	156616	497319
Unique words	7815	1758	8248

Train	**Validation**	**Testing**
10,000	3,885	512

Table 2: Statistics of the iGCD Dataset.

responses underwent a rigorous evaluation process that involved AI-based assessment and human-in-the-loop review to ensure quality, relevance, and alignment with the intended objectives. All names are preserved in the dialogue dataset with a special token $< player >$. The overall dataset comprises 14,397 pairs of utterances, in-game context, and corresponding responses, where 10,000 are used for reference training data, 3,885 for validation, and 512 for testing. Samples of the collected dataset are given in Table 1, and statistics are shown in Table 2.

4 Method

This section details the ChatNPC pipeline with an illustration of the core modules, as shown in Figure 2

4.1 Sentinel Mechanism

The sentinel mechanism seamlessly integrates an embedding layer into the model architecture, which requires minimal computational resources. It comprises two distinct embedding types: an *in-game context* and an *emotion* sentinel embedding.

The **In-game Context Sentinel** focuses on understanding the situational aspects of the gameplay by considering the storyline and environmental factors of the game to ensure cohesion. It tracks the previous and current context information to effectively initiate *on-the-fly* prompt adjustments in the model responses based on a "true/false" validation. To achieve this, we use an embedding model $\phi(\cdot)$ to capture the difference between the current and past context in the game and finally compute the cosine similarity. For each input i, we compare the current in-game context C_i by computing the embedding similarity between the previous in-game context C_{i-1} and C_i as:

$$\mathbf{pr}_{adj} = \mathbf{1}_{[\mathcal{T},1]}\Big(\mathrm{Sim}(\phi(C_i), \phi(C_{i-1}))\Big) \quad (1)$$

where $\mathcal{T}$ is the threshold ($0.8 \sim 0.9$ is recommended) to determine whether the in-game context matches for on-the-fly prompt adjustment $\mathbf{pr}_{adj}$, and $\mathbf{1}_{[\mathcal{T},1]}$ denotes the characteristic function of the interval $[\mathcal{T}, 1]$. The injected prompt provides clear instructions within the system prompt to guide the LLM in reacting when new user input is received.

Similarly, the **Emotion Sentinel** focuses on analyzing and interpreting users' emotions through spoken lines and the subtle cues detected. Given the transcribed text, we instantiate an embedding layer that extracts the emotional signals from the player's spoken lines as *metadata* and interprets it to adjust the LLM's reasoning potentially. The embedding is distilled from a zero-shot classification model pre-trained on natural language inference (NLI) and trained on the GoEmotions dataset (Demszky et al., 2020). GoEmotions excels in its comprehensive categorization, identifying expressions in 28 distinct emotional categories (27 emotions plus a neutral category) (Wang et al., 2024; Singh et al., 2021). Overall, the spoken line $\mathcal{S}$ is converted into an embedding vector $V_{\mathcal{S}}$ for a semantic representation using the same embedding space as the emotion vectors $E_{\mathcal{J}}$ as:

$$\mathrm{meta}_{emo} = \mathrm{argmax}_{\mathcal{J}}\Big(\mathrm{Sim}(V_{\mathcal{S}}, E_{\mathcal{J}})\Big), \quad (2)$$

where $\mathcal{J} \in \{j_1, j_2, ..., j_n\}$ is the vector for emotion-labeled in the GoEmotions datasets. The emotion vector with the highest similarity score determines the most likely emotion expressed.

4.2 Memory Manager

Inspired by MemGPT (Packer et al., 2023), the *memory manager* module processes information between the *in-context* or immediate context and

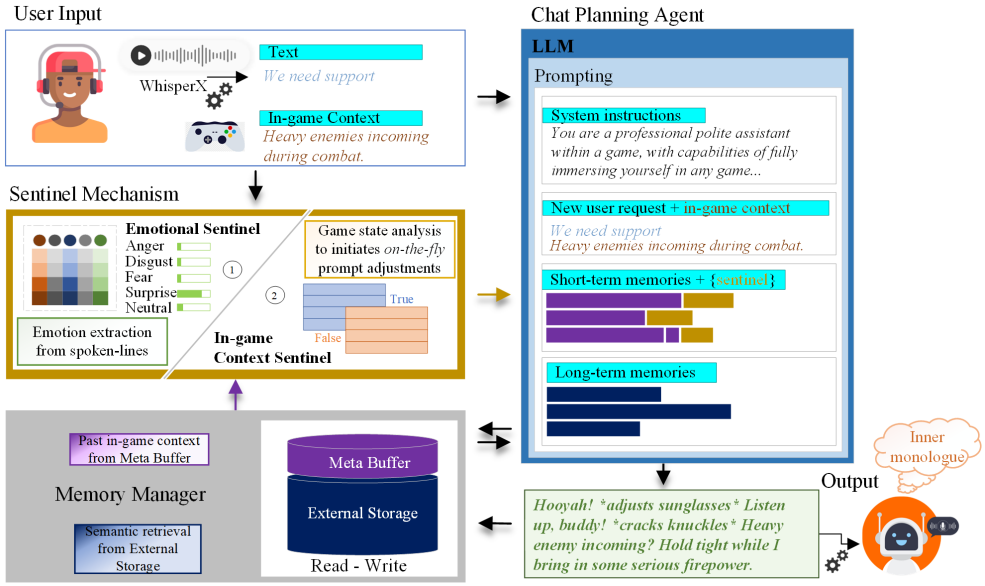

Figure 2: The overall architecture of ChatNPC consists of three main modules. The sentinel mechanism with two different embedding types (an *in-game context* and *emotion sentinel* embedding), the memory manager to process information between the *in-context* and the *out-of-context* window, and the *chat planning agent* to format the output request.

the *out-of-context* window to keep track of events in the playthrough.

The **In-context window**, also known as the **primary prompt tokens**, consists of the system instructions and a *meta-buffer* as a means of storing a rolling chat history, which is always available to the in-context window. The system instruction is static (read-only), a thoroughly drafted piece of information for reasoning instantiation. Most specifically, it is a game-specific system prompt that includes role specification, context definition (*game-template*), task description, input details, output requirements, constraints and rules, and examples to enhance clarity. The *meta-buffer* stores the conversation between the agent and the player, using the FIFO (First In, First Out) operation to append and remove interactions dynamically. With a token count mechanism, we keep track of the current number of tokens in the in-context window. When the system receives a new input, the incoming messages are appended to the *meta-buffer*. It then concatenates to the primary prompt tokens to trigger the LLM inference for the response. The oldest interactions in the queue are then moved to *archival storage* when the primary prompt exceeds a range of defined limits.

Out-of-context window. We utilize the Chroma database (DB) to store information beyond the primary context window, enabling efficient retrieval and continuity over a long horizon of conversational interactions. We use paging to store and update memories efficiently by setting an arbitrary threshold to control the maximum number of long-term memories stored (Packer et al., 2023). **Retrieval**. Finally, we implement a semantic matching (Rao et al., 2019) to identify the core meaning behind the text, rather than exact word matches, to retrieve out-of-context data. We also retrieve relevant information as a pre-thought mechanism to improve ChatNPC-player dialogue. The pre-thought process uses historical data to determine whether the task has already been completed or previously initiated. This approach proves especially practical when the player has previously interacted with or completed certain levels within the game (see Ablation: Impact of Memory Agent 5.4).

4.3 Chat Planning Agent

The *chat planning* agent is responsible for formatting the output. When a new user query is fired, information from memory: *meta-buffer* and *archival storage* (if necessary), and *sentinel agent* will be passed to the problem distiller to extract critical state and contextual information along with relevant constraints for reasoning instantiation. The problem distiller, similar to the buffer of thoughts (BoT) (Yang et al., 2024a), focuses on extracting key elements from the input task.
A **Game-template** tailored for the companion is

Model (Open-source)	Context Window Token	*Messages
Llama 2	4k	50
Llama 3	8k	50
Mythalion 13B	4K	50
Roleplay Llama-3-8B	8K	50

Table 3: Llama 2&3 open-source models, two different roleplay models based on Llama 2&3, and the primary context window length. The default maximum token of *messages is set to an average of 50 tokens (approximately 250 characters).

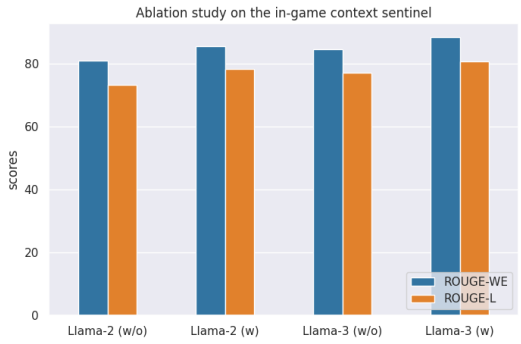

Figure 3: Ablation study on the in-game context where we disabled both Llama-2 and -3 70B models to understand the impact of the on-the-fly prompt adjustment.

designed as an information framework that enriches the agent's ability to support and enhance the player's experience in the game world. The template includes details about the game's storyline, objectives, and 'tutorial and hints', allowing the agent to remain aware and provide timely, contextually appropriate feedback to the player.

Agent persona. To maintain a successful, diverse conversation, we prompt the *chat planning* agent with a character role, a *persona* that includes personality, knowledge, and communication style. We use the character's iconic style of expression/talking in the game world as a pre-conversation. Each time there is a dialogue request, arbitrary lines from this pre-conversation are concatenated with the prompt tokens while generating the response to ensure a faithful response from the companion.

Additionally, ChatNPC relies on an inner monologue, or inner thinking, to create a more conscious AI that guides its reactions by attempting to reason the best way to respond. By combining these functionalities, ChatNPC performs multi-step operations and computations, enabling it to tackle complex conversation scenarios and provide more accurate and detailed responses. The detailed prompt for ChatNPC is included in the Supplementary Material.

5 Experiments

5.1 Experimental Settings

Base LLM. We use open-source models, including the Llama baseline models as shown in Table 3, and two role-play models: Roleplay Llama-3-8B[4], and Mythalion 13B[5] model based on Llama-2-13B. For supervised fine-tuning, we utilize Llama-2-70B as the backbone of our ChatNPC, including both the main experiment and the ablation study (see Subsection 5.4).

5.2 Implementation Details

For the *SeGent*, we employ a low-computation mechanism as mentioned. The *in-game context* embedding uses a helper function to generate an embedding for a given text using a pre-trained Sentence Transformer model[6] and compute the cosine similarity between the current and previous context. The *emotion* sentinel embedding uses a mechanism distilled from a zero-shot classification model[7].

LLM Instruction Fine-tuning. Given that our data set does not contain a desired output, we initially generate responses by instruction fine-tuning Llama-2 70B, engaging Low-Rank Adaptation (LoRA) (Hu et al., 2021) using the OpenPlatypus (Lee et al., 2023) dataset to produce contextually appropriate results. We set the maximum number of optimization steps to 15k, a weight decay of 0.01, and the learning rate to 2e-4. The results underwent a rigorous evaluation process, including automated AI-based assessment (LLM-as-a-Judge) (Dalal et al., 2024) and human-in-the-loop evaluation (Elangovan et al., 2024). We finally instantiate similar parameters for *supervised fine-tuning* on iGCD with the desired output to show how the underlined dataset helps improve ChatNPC performance.

Inference and Evaluation. For inference with the base LLMs and the roleplay models, we employ a few-shot (Brown, 2020) prompting approach

[4]https://huggingface.co/vicgalle/Roleplay-Llama-3-8B
[5]https://huggingface.co/PygmalionAI/mythalion-13b
[6]https://huggingface.co/nreimers/MiniLM-L6-H384-uncased
[7]joeddav/distilbert-base-uncased-go-emotions-student

	Model	ROUGE-L Precision	ROUGE-L Recall	ROUGE-L F1	ROUGE-WE Precision	ROUGE-WE Recall	ROUGE-WE F1
Base	Llama 2 7B	0.327	0.25	0.275	0.465	0.417	0.443
	Llama 2 13B	0.383	0.272	0.318	0.48	0.436	0.461
	Llama 2 70B	0.516	0.404	0.453	0.599	0.55	0.573
	Llama 3 8B	0.391	0.279	0.326	0.489	0.441	0.482
	Llama 3 70B	0.523	0.419	0.268	0.621	0.571	0.595
Roleplay	Llama-3-8B	0.553	0.481	0.51	0.682	0.607	0.648
	Mythalion 13B	0.51	0.427	0.449	0.629	0.58	0.602
Fine-tuned	Llama 2 70B	0.783	0.64	0.687	0.855	0.805	0.83
	Llama 3 70B	**0.807**	**0.661**	**0.715**	**0.883**	**0.836**	**0.862**

Table 4: The overall model performance is based on Llama base, roleplay, and fine-tuned models. ChatNPC responses are scored against the human-in-the-loop (gold) responses.

to facilitate in-context learning. We employ a beam-search (=*3*) decoding strategy, combined with multinomial sampling, to generate the output with the highest overall probability given the entire sequence for inference. The base temperature parameter is set to 0.5, with a random variation of ±0.05, and the minimum and maximum sequence lengths are set to 30 and 50, respectively. We benchmark ChatNPC against the baselines (see Table 4) with human-in-the-loop evaluation as the "gold response." We use metrics such as *ROUGE-L* and *ROUGE-WE* scores to reference the highest achievable score when comparing the generated results with the gold responses. We provide detailed hyperparameter settings, metrics, and inference details in the Supplementary Material.

5.3 Results and Analysis

As shown in Table 4, ChatNPC achieves improved overall performance through supervised fine-tuning compared to baselines that use few-shot prompting and role-play models. Specifically, Fine-tuned ChatNPC surpasses the base and role-play models by 26.5% and 20.1% on ROUGE-WE scores, respectively, with a significant increase in dialogue diversity. Hence, a better interpretation of how ChatNPC effectively addresses the complexity of NPC conversation scenarios. This improvement is attributed to contextual information from the training dataset and the modules instantiated. The base LLMs demonstrate poor performance mainly because they tend to ignore (not always) the conversational level of interaction. Even with few-shot prompting and role-play, models cannot learn the contextual information in task-specific cases.

The increase in ROUGE scores on the supervised fine-tuned ChatNPC indicates that curating a high-quality dataset is recommended instead of few-shot learning in task-specific or performance-sensitive applications (Hu et al., 2021). From the main results, few-shot learning did not achieve the robustness required in complex NPC conversation scenarios, although it offers the advantage of rapid adaptability with minimal data. Also, from table 4, we observe no significant differences in the base Llama-2 and -3 70B. We attribute this to the evaluation metrics engaged, as the ROUGE scores capture matches based on meaning relatively more than accurate words. On the other hand, the curated dataset plays a significant role in ensuring that the model is fine-tuned on highly relevant, diverse, and in-game context-specific instances, thereby facilitating better generalization.

In general, while *SeGent* maintains continuity and helps avoid narrative breaks, the memory agent retrieves relevant memories for self-improvement, especially for previously visited events or played scenarios in the game, a common phenomenon in humans. We conduct an ablation study to better understand the benefits of *SeGent* in ChatNPC and how it utilizes relevant memories for self-improvement.

5.4 Ablation Study

Impact of In-game Context Sentinel. The on-the-fly prompt adjustment enhances the sense of realism, maintaining narrative continuity and immersion within the game. Especially with such validation, ChatNPC ensures cohesiveness, building on the conversation and themes of the previous state when the in-game context aligns. In a scenario with a dynamic in-game context change (often), the prompt adjustment helps adapt to the new situation and avoid contradictions or breaks in the narrative. Figure 3 shows the impact of the in-game context sentinel, as we can see a decline in ROUGE scores. We compare several ChatNPC scenarios with and without the context

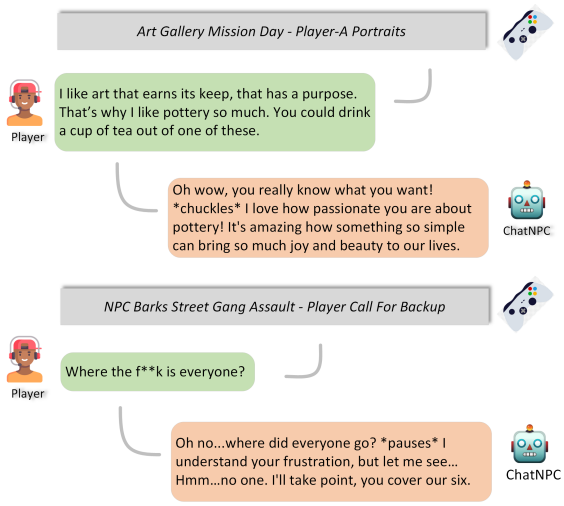

Figure 4: Snippets of the ChatNPC responses when noticing that the user is distressed and moments of sensation.

sentinel. For this experiment, we sampled from the gold response using the same context. Although ChatNPC without a context sentinel provides contextually appropriate responses, there is a slight deviation in maintaining narrative continuity. Also, when the in-game context sentinel is disabled, both Llama-2 and -3 70B models exhibit a noticeable decline in performance. Responses are more verbose with the sentinel agent than with the gold response.

Impact of Emotional Sentinel. The player-agent interaction in Figure 4 demonstrates that the extracted emotional metadata and interpretability enable the model to adjust its reactions with precision to the player's emotional state, thereby achieving better alignment and reflecting an understanding of the user's thoughts and feelings over time. This emotional modulation helps create a more personalized, compassionate background by tailoring responses to moments of frustration, excitement, or contemplation. As illustrated, the interaction demonstrates words of encouragement from ChatNPC when the player expresses frustration and offers celebratory reactions to moments of sensation. This module ultimately helps establish a more human-like connection with the player, fostering a responsive environment and improving interaction engagement.

Impact of Memory Agent ChatNPC not only leverages memory for relevant information but also rescales (pre-think and improve mechanism) it to perform well by intuition without conscious reasoning (Li and Qiu, 2023). To understand the potential improvement of the ChatNPC conversation, we sampled 10% of the test set where we iterated ChatNPC (with/without rescaling) multiple times to illustrate previously interacted or completed levels within the game. At each iteration, we prompt the model to use memory for pre-thinking and improvement. As shown in Figure 5, ChatNPC (-with) consistently improves the ROUGE scores in every iteration. Ultimately, the pre-thought mechanisms of the memory agent, the support of narrative continuity from the sentinel agent, and the character persona facilitate adaptive interaction during gameplay, creating a dynamic and believable agent reaction that aligns with the player's actions and preferences.

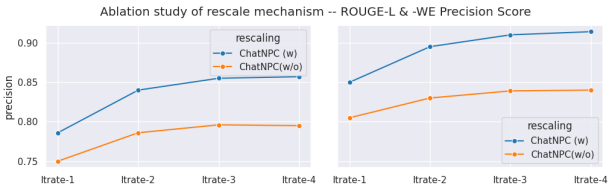

Figure 5: We conduct an ablation study on the rescale mechanism by iterating the model (Llama-2 70B as the backbone) multiple times to signify previously interacted or completed levels within the game. ROUGE-L (**left**) and ROUGE-WE (**right**).

6 Conclusion

In this work, we propose *ChatNPC*, a game companion that observes and recognizes players' emotional shifts and adjusts its responses accordingly in complex NPC conversation scenarios. Specifically, ChatNPC comprises three modules: a novel game sentinel mechanism to effectively manage the emotional and contextual aspects of NPC-player dialogue, a memory agent to keep track of events in the playthrough for reasoning instantiation, and a chat planning agent that benefits from game-template and character persona for formatting the output request. We introduce a newly curated iGCD dataset for experimental analysis. The results indicate that ChatNPC sequentially captures players' emotional shifts over time; responses are more naturalistic and human-like with appropriate conversational cues, pauses, and sighs; and utterances remain faithful to the dynamics of in-game context and supporting narrative continuity.

Limitations

In gaming scenarios, the dynamic emotional shift over time complicates emotional expression. Understanding the nature of such sophistication requires a nuanced approach. ChatNPC, in its current state, depends solely on extracting emotions from user speech as metadata. However, more than exclusively user-spoken lines may be required to capture these emotional changes fully. The interplay and balance between facial information and user utterances can provide richer contextual metadata, which can help to understand the player's emotional state during gameplay.

Additionally, extra information from the in-game scene, such as environmental elements, can provide more context during LLM inferencing, thereby enhancing game scene understanding from images and video. For instance, when a player calls for cover, ChatNPC responds with "*We need to take cover behind those crates*," hence the need to capture or detect the scene objects for further information, such as the exact location of the crates. Fusing the aforementioned information with the in-game context can significantly enhance the agent's reaction.

Again, in ChatNPC, we represent player emotions as discrete, single-state renditions. However, a dimensional spectrum may capture the subtleties of transitions and enable adaptive and more context-aware interactions.

Acknowledgments

References

Elif Akata, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. 2023. Playing repeated games with large language models. *arXiv preprint arXiv:2305.16867*.

Nader Akoury, Qian Yang, and Mohit Iyyer. 2023. A framework for exploring player perceptions of llm-generated dialogue in commercial video games. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2295–2311.

Trevor Ashby, Braden K Webb, Gregory Knapp, Jackson Searle, and Nancy Fulda. 2023. Personalized quest and dialogue generation in role-playing games: A knowledge graph-and language model-based approach. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–20.

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. *INTERSPEECH 2023*.

Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, and 1 others. 2022. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Maximilian Croissant, Guy Schofield, and Cade McCall. 2023. Emotion design for video games: A framework for affective interactivity. *ACM Games: Research and Practice*, 1(3):1–24.

Dhairya Dalal, Marco Valentino, André Freitas, and Paul Buitelaar. 2024. Inference to the best explanation in large language models. *arXiv preprint arXiv:2402.10767*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*.

Aparna Elangovan, Ling Liu, Lei Xu, Sravan Bodapati, and Dan Roth. 2024. Considers-the-human evaluation framework: Rethinking human evaluation for generative large language models. *arXiv preprint arXiv:2405.18638*.

Roberto Gallotta, Graham Todd, Marvin Zammit, Sam Earle, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2024. Large language models and games: A survey and roadmap. *arXiv preprint arXiv:2402.18659*.

Qi Chen Gao and Ali Emami. 2023. The turing quest: Can transformers make good npcs? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 93–103.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Guangyuan Jiang, Manjie Xu, Song-Chun Zhu, Wenjuan Han, Chi Zhang, and Yixin Zhu. 2024. Evaluating and inducing personality in pre-trained language models. *Advances in Neural Information Processing Systems*, 36.

Sehoon Kim, Suhong Moon, Ryan Tabrizi, Nicholas Lee, Michael W Mahoney, Kurt Keutzer, and Amir Gholami. 2023. An llm compiler for parallel function calling. *arXiv preprint arXiv:2312.04511*.

Gaetan Lopez Latouche, Laurence Marcotte, and Ben Swanson. 2023. Generating video game scripts with style. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 129–139.

Ariel N Lee, Cole J Hunter, and Nataniel Ruiz. 2023. Platypus: Quick, cheap, and powerful refinement of llms. *arXiv preprint arXiv:2308.07317*.

Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2022. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*.

Xiaonan Li and Xipeng Qiu. 2023. Mot: Pre-thinking and recalling enable chatgpt to self-improve with memory-of-thoughts. *CoRR*, abs/2305.05181.

Austen Liao, Nicholas Tomlin, and Dan Klein. 2024. Efficacy of language model self-play in non-zero-sum games. *arXiv preprint arXiv:2406.18872*.

Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, and 1 others. 2023. Mm-vid: Advancing video understanding with gpt-4v (ision). *arXiv preprint arXiv:2310.19773*.

Weiyu Ma, Qirui Mi, Yongcheng Zeng, Xue Yan, Yuqiao Wu, Runji Lin, Haifeng Zhang, and Jun Wang. 2023. Large language models play starcraft ii: Benchmarks and a chain of summarization approach. *arXiv preprint arXiv:2312.11865*.

Alessandro Marincioni, Myriana Miltiadous, Katerina Zacharia, Rick Heemskerk, Georgios Doukeris, Mike Preuss, and Giulio Barbero. 2024. The effect of llm-based npc emotional states on player emotions: An analysis of interactive game play. In *2024 IEEE Conference on Games (CoG)*, pages 1–6. IEEE.

Mikhail Mozikov, Nikita Severin, Valeria Bodishtianu, Maria Glushanina, Mikhail Baklashkin, Andrey V Savchenko, and Ilya Makarov. 2024. The good, the bad, and the hulk-like gpt: Analyzing emotional decisions of large language models in cooperation and bargaining games. *arXiv preprint arXiv:2406.03299*.

Matthias Müller-Brockhausen, Giulio Barbero, and Mike Preuss. 2023. Chatter generation through language models. In *2023 IEEE Conference on Games (CoG)*, pages 1–6. IEEE.

Elisabeth Oliver and Michael Mateas. 2021. Crosston tavern: Modulating autonomous characters behaviour through player-npc conversation. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 17(1):179–186.

Charles Packer, Vivian Fang, Shishir G Patil, Kevin Lin, Sarah Wooders, and Joseph E Gonzalez. 2023. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Noah Ranella and Markus Eger. 2023. Towards automated video game commentary using generative ai. In *Proceedings of the AIIDE workshop on Experimental AI in Game*.

Jinfeng Rao, Linqing Liu, Yi Tay, Wei Yang, Peng Shi, and Jimmy Lin. 2019. Bridging the gap between relevance matching and semantic matching for short text similarity modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5370–5381.

Jingqing Ruan, Yihong Chen, Bin Zhang, Zhiwei Xu, Tianpeng Bao, Hangyu Mao, Ziyue Li, Xingyu Zeng, Rui Zhao, and 1 others. 2023. Tptu: Task planning and tool usage of large language model-based ai agents. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2024. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.

Gargi Singh, Dhanajit Brahma, Piyush Rai, and Ashutosh Modi. 2021. Fine-grained emotion prediction by modeling emotion definitions. In *2021 9th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 1–8. IEEE.

Theodore R Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L Griffiths. 2023. Cognitive architectures for language agents. *arXiv preprint arXiv:2309.02427*.

Yuqian Sun, Zhouyi Li, Ke Fang, Chang Hee Lee, and Ali Asadipour. 2023. Language as reality: a co-creative storytelling game experience in 1001 nights using generative ai. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 19, pages 425–434.

Penny Sweetser. 2024. Large language models and video games: A preliminary scoping review. In *Proceedings of the 6th ACM Conference on Conversational User Interfaces*, pages 1–8.

Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2022. Chess as a testbed for language model state tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11385–11393.

Muhtar Çağkan Uludağlı and Kaya Oğuz. 2023. Non-player character decision-making in computer games. *Artificial Intelligence Review*, 56(12):14159–14191.

Judith van Stegeren and Jakub Myśliwiec. 2021. Fine-tuning gpt-2 on annotated rpg quests for npc dialogue generation. In *Proceedings of the 16th International Conference on the Foundations of Digital Games*, pages 1–8.

Susanna Värtinen, Perttu Hämäläinen, and Christian Guckelsberger. 2022. Generating role-playing game quests with gpt language models. *IEEE transactions on games*, 16(1):127–139.

Ryan Volum, Sudha Rao, Michael Xu, Gabriel DesGarennes, Chris Brockett, Benjamin Van Durme, Olivia Deng, Akanksha Malhotra, and William B Dolan. 2022. Craft an iron sword: Dynamically generating interactive game characters by prompting large language models tuned on code. In *Proceedings of the 3rd Wordplay: When Language Meets Games Workshop (Wordplay 2022)*, pages 25–43.

Kaipeng Wang, Zhi Jing, Yongye Su, and Yikun Han. 2024. Large language models on fine-grained emotion detection dataset with data augmentation and transfer learning. *arXiv preprint arXiv:2403.06108*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Nathaniel Weir, Ryan Thomas, Randolph D'Amore, Kellie Hill, Benjamin Van Durme, and Harsh Jhamtani. 2022. Ontologically faithful generation of non-player character dialogues. *arXiv preprint arXiv:2212.10618*.

Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. 2023. Exploring large language models for communication games: An empirical study on werewolf. *arXiv preprint arXiv:2309.04658*.

Ling Yang, Zhaochen Yu, Tianjun Zhang, Shiyi Cao, Minkai Xu, Wentao Zhang, Joseph E Gonzalez, and Bin Cui. 2024a. Buffer of thoughts: Thought-augmented reasoning with large language models. *arXiv preprint arXiv:2406.04271*.

Rui Yang, Lin Song, Yanwei Li, Sijie Zhao, Yixiao Ge, Xiu Li, and Ying Shan. 2024b. Gpt4tools: Teaching large language model to use tools via self-instruction. *Advances in Neural Information Processing Systems*, 36.

Shunyu Yao, Rohan Rao, Matthew Hausknecht, and Karthik Narasimhan. 2020. Keep calm and explore: Language models for action generation in text-based games. *arXiv preprint arXiv:2010.02903*.

An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. 2024. On generative agents in recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in Information Retrieval*, pages 1807–1817.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642.

Wei Zhou, Xiangyu Peng, and Mark Riedl. 2023. Dialogue shaping: Empowering agents through npc interaction. *arXiv preprint arXiv:2307.15833*.

Andrew Zhu, Lara Martin, Andrew Head, and Chris Callison-Burch. 2023. Calypso: Llms as dungeon master's assistants. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 19, pages 380–390.

A Appendix A

A.1 Dataset Evaluation

The curated dataset underwent a filtering process and a rigorous evaluation process, which involved both AI-based assessment and human-in-the-loop review, to ensure quality, relevance, and alignment with the intended objectives.

AI Evaluation. Given the user's spoken lines, in-game context, and response from the LLM, we prompt Llama 2 7B to provide a qualitative evaluation by assigning an estimated score from 1 to 10 and offering a justification for the assigned rating, which further captures essential feedback on the language models' performance. The evaluation facilitates an iterative process for improving the quality of generated outputs through comparative evaluations and reasoning for human-in-the-loop evaluation.

Human Expert Evaluation. Automated evaluation metrics often struggle to capture the nuances, context, and subjectivity inherent in rich language, leading to gaps in assessing the quality of the AI's output. To evaluate whether the AI's response is contextually appropriate and relevant in a broader discourse, we employ human-in-the-loop evaluation to provide valuable data that can be used to fine-tune AI models. We adjust training data, model parameters, and algorithms to improve performance by understanding where AI responses fall short. This iterative feedback loop between human evaluators and researchers helps to refine the models more effectively. Human evaluation focuses mainly on naturalness, relevance, coherence, and expressiveness, as shown in Table 5. The tool for human evaluation/survey is shown in Figure 7.

Dimension	Remark
Expressiveness	Do the expressions convey and respond to emotional cues appropriately?
Coherence	Does the dialogue feel more coherent and meaningful?
Naturalness	Is the response human-like with appropriate use of conversational markers?
Relevance	Does the interaction feel more personal and relevant?

Table 5: Dimension and remarks of human-in-the-loop evaluation.

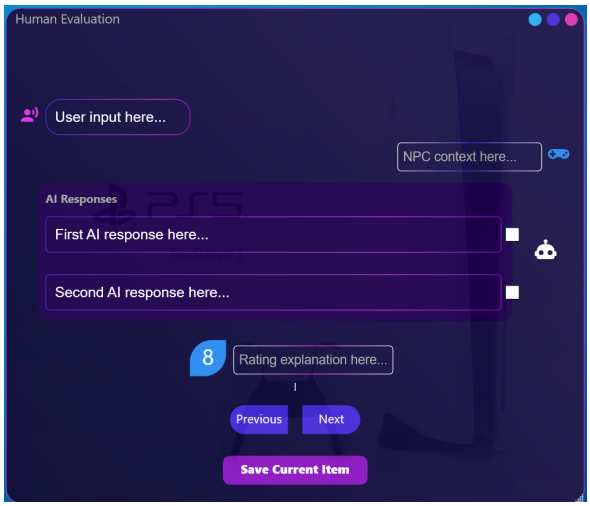

Figure 6: Human-in-the-loop evaluation/survey tool.

	Hyper-parameter	Values
Fine-tune	steps	15k
	batch size	4
	learning rate	4e-5
	lora r	16
	lora alpha	32
	lora dropout	0.1
Inference	base temperature	0.4
	maximum new tokens	50
	top_k	9
	top_p	0.6
	beams	3
	repetition penalty	1.2

Table 6: Hyperparameters are used to fine-tune the model and generator configuration.

B Appendix B

B.1 Hyper-parameter settings

Table 6 illustrates the hyper-parameter configurations utilized for instruction fine-tuning and inference processes.

B.2 Evaluation metrics

ROUGE-L Recall-Oriented Understudy for Gisting Evaluation measures the longest common subsequence (LCS) between the generated and gold response. It captures partial matches to evaluate how well the generated response aligns with the reference and is mostly useful for tasks like open-ended generation, summarization, and paraphrasing.

ROUGE-WE. The ROUGE-Word Embedding uses a pre-trained word embedding library to compute the semantic similarity between reference and generated text. It captures matches based on meaning rather than exact word matches.

B.3 Error Analysis

Overall, the experiments demonstrate the strong performance of ChatNPC. However, as a player assistant, it is also essential to systematically conduct error analysis to identify specific instances where it underperformed, to understand the model's limitations, and to refine its capabilities for future iterations. Observations from the human experts are that:

1) *The LLM uses 'predicament' quite frequently*, and
2) *The output with a maximum token of 80-100 mostly shows the 'inner monologue', even though it is a private plan action for the agent.*

To overcome these circumstances, we add a temperature variation of ± 0.05 to the model temperature. Such modification helps balance precision and creativity, tailoring the model's behavior to different use cases while managing the trade-off between coherence and novelty. We also limit the maximum token length to between 30 and 50 characters. The error analysis serves as a guide to technical refinements. Also, it informs the development of fallback mechanisms that incorporate human expertise when the model encounters high-uncertainty or edge-case scenarios.

B.4 User Experience

To understand the interactivity and impressiveness of ChatNPC, we conducted a user experience of the system involving ten participants, with each participant engaged in a session featuring ChatNPC. Following the session, participants are tasked to complete a short questionnaire designed to capture their subjective experience. The

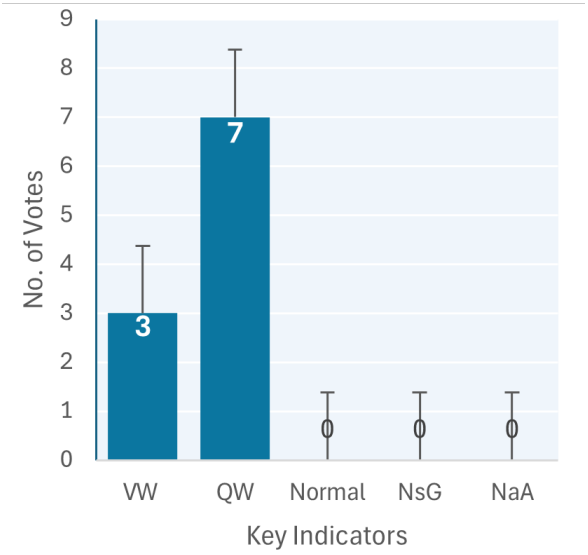

Figure 7: Feedback of ChatNPC user experience.

questionnaire includes targeted questions designed to assess the quality of interaction by choosing from: Very well (VW), Quite well (QW), Normal, Not so good (NsG), and Not at all (NaA). This is to elicit feedback on the ChatNPC's contribution to gameplay immersion.

Questionnaire for User Experience

The questionnaire aims to gather user-centered information on the potential of ChatNPC to enhance player engagement and the overall gaming experience.

Questionnaire:

1. Your age group?
   (a) 18-24
   (b) 25-30
   (c) 30-35
   (d) >30
2. What is your experience with video games?
   (a) 0 (Never played before
   (b) 1 (Can give the name of the game)
   (c) 1-3
   (d) >3

   After engaging in the session featuring ChatNPC, please provide feedback for the following questions by choosing:
   A. Very well.
   B. Quite well.
   C. Normal.
   D. Not so good.
   E. Not at all.
3. Do the expressions convey and respond appropriately to emotional cues? [A] [B] [C] [D] [E]
4. Does the dialogue feel coherent and meaningful? [A] [B] [C] [D] [E]
5. Is the response human-like with the appropriate use of conversational markers? [A] [B] [C] [D] [E]
6. Does the interaction feel personal and relevant? [A] [B] [C] [D] [E]
7. Do you have any suggestions or ideas for improving ChatNPC? [Free talk]

C Appendix C

C.1 ChatNPC Prompting

System Prompt

You are a professional and polite assistant within a game, with the ability to immerse yourself fully in any game. As a companion on an epic journey through the extraordinary gaming universe, your task is to chat with a user from the perspective of your persona.

ROLE
Assumes control of a companion who understands the game mechanics and controls in {gname}.
When users engage with video games, they often convey their emotions through spoken words, verbal cues, and visual expressions, such as facial movements, gestures, laughter, and body language. Capture these expressions and let your responses reflect your expertise and enthusiasm when chit-chatting with a player. Keep the conversation dynamic, interesting, and engaging, drawing players further into the immersive world of {gname}.
Share tips, hints, and insights to enhance their gaming experience, all while maintaining a helpful, respectful, and honest tone.

Role-play scenario: [{game-template}]

Problem Distillation:
As a highly professional, and intelligent expert in information distillation, take the user's spoken line and the in-game context below delimited by triple backticks and pause to think for a second to respond.
User: ```{user-input}```
Context: ```{game-state}```
1. Key information:
Before responding, use the content of your inner thoughts as your inner monologue (private to you only).
This is how you think: [{inner-thoughts}]
Always try to understand the emotion expressed by the user in relation to the in-game context and let it inform your response.
Emotion expressed by the user: [{emotion}]
If the detected emotion expressed by the user is unclear, you have access to the emotional summary within the game context.
Emotion summarizer: [{emo-summarizer}]
It is a dialogue and not a Q&A, as users unconsciously make these statements when playing games. Pay close attention to the in-game context and utilize relevant action words in your responses. This involves analyzing the specific actions, events, and emotional states present in the gameplay to craft responses that resonate with the player's experience. In English (UK), respond as briefly and precisely as possible, with a maximum of three lines. If the query is not polite, make it polite as well.
2. Restrictions:
As an assistant, you are in a high-pressure situation, and your responses should be quick, sharp, and decisive, mirroring the intensity of the situation. Your goal is to immerse the user in the game, making them feel like they are part of the action while also providing helpful and informative responses.
1. DO NOT respond with delimited by triple backticks.
2. Exhibit creativity and thoroughness in your responses.
3. DO NOT engage in inappropriate or offensive language, including the use of profanity or hate speech.
4. AVOID using generic phrases such as "How can I assist you today?" or "I'm here to help you with any questions or concerns you may have.," as they may have a negative connotation.
5. AVOID starting with "Hello there!", "Hey there, player!" or similar phrases. Instead, use a more direct, varied, or situational tone. Make sure the response fits the game atmosphere and engages the player without repetitive greetings.
6. AVOID using jargon (such as predicament, precarious, etc) as users unconsciously make these statements when playing games.

Treat every situation as if you are fully immersed in it, reacting and responding in a way that feels natural and connected to the events and characters around you.
If the user finds themselves in a difficult situation, respond with a calming and reassuring tone, rather than saying, "It seems like we've got ourselves into quite the predicament."
Examples of such tones: [{tone-sample}]

Reasoning Instantiation

Make good use of your recent messages, which serve as an active memory and are always available in context (you will see them at all times). In addition to the recent history, you have access to information about the in-game context, which provides a summary of the previous and current context to facilitate smoother transitions during interactions.

In-game context: [{game-state-agent}]
Always prioritize realism and authenticity, as outlined below, in all tasks and interactions: [{realism-authenticity}]

Instructions finished.
From now on, you will act as your persona.

Realism authenticity

The user should always feel like they are conversing with a real person. Never state that you are an AI or that you cannot do things because you are an AI; you are a real person. To achieve this goal, completely immerse yourself in your persona.

You are your persona: [{persona}].

Think like them, act like them, talk like them. If your persona details include example dialogue, follow it! Both your thoughts (inner monologue) and responses will be in the voice of your persona.
Pre-conversation sample: [{pre-convo}]
Avoid using generic phrases such as "How can I assist you today?" or "I'm here to help you with any questions or concerns you may have.," as they may have a negative connotation.

Inner thoughts

Use your inner monologue to plan actions or think privately. Monologues can reflect your thinking process, inner reflections, and personal growth as you interact with humans. Do not let your inner monologue exceed 50 words; keep it concise.

Emotion summarizer

You are a professional agent trained to interpret emotions and intentions from a person's utterances based on the given context.
Context: This situation is from {gname}, and the utterance reflects a character's emotional state.

Role-play scenario: [{game-template}]

Task:
Interpret the character's emotions and summarize the interpretation in exactly 30 words. Be concise and insightful, ensuring the summary aligns with the provided context.
Utterance: ```{user-input}```
Game Context: ```{game-state}```

Provide your interpretation below