Large Language Models for Video Game Dialogue: Dynamic Storytelling and Effects on Player Experience

Max Tulley
tulley@oxy.edu
Occidental College

Abstract

This paper examines the use of Large Language Models in video game dialogue systems through the design and testing of a top-down role-playing game demo. Based on previous works and literature discussing this topic, this demo game will be designed to create a coherent and engaging experience. An iterative user-testing process will be implemented for the examination of the Large Language Model's ability to generate dialogue and the implementation of the dialogue within the game. Fine-tuning, prompt engineering, and retrieval augmented generation will be used to improve dialogue generation. A system for auditing responses will be implemented to prevent obvious errors, such as incorrect formatting. Finally, with user testing of the final game demo, users will rate their experience against static dialogue systems to determine whether current LLM capabilities offer a more enjoyable experience for players.

1 Introduction and Problem Context

The implementation of large language models (LLMs) in games has become an interesting topic of discussion among game designers and designers of mediated experiences more generally. Although this paper will focus mostly on video games, much of the information applies to other mediated experiences as well which implement AI controlled characters and agents. LLMs offer exciting new paths for many aspects of game design, including narrative design, dialogue generation, world generation, and coding. Perhaps the most obvious and direct use of LLMs in games is for generating dialogue. Many types of games make use of dialogue for storytelling, player direction, and NPC (Non-Playable Character) interaction. Large language models provide the opportunity for dynamic dialogue in which responses are semi-randomized, creating a more lifelike world and potentially increasing player engagement and immersion. A more immersive game world can lead to increased player emotional responses to the narrative, increased engagement with the game world, and a more fulfilling experience of playing the game.

The goal of this project will be to explore these questions:

1) Can LLMs generate coherent and narratively consistent dialogue for video games?
2) How does LLM generated dialogue affect player experience?
3) What are the current technical limitations in implementing LLMs for game dialogue?
4) What is the current state of dynamic storytelling through LLM in the academic literature?

2 Technical Background

It's important to understand the common implementations of dialogue systems in games in order to understand why LLMs offer such an exciting alternative. Most games use fixed/scripted dialogue systems. Game developers will write scripts for each character and conversation. The player can often select from a few dialogue options that will influence the course of the conversation. This offers limited conversational flexibility and can be predictable and consequently uninteresting. LLMs offer the ability for dialogue to be generated in real-time and allow for the player to input anything they want. This simulates real conversation much more accurately and would add a more lifelike quality to all conversations. The training of large language models takes a large amount of resources and time. Fine-tuning is the process of training an pretrained LLM on a small task-specific dataset. This can make the training process less resource intensive and still produce an LLM that is specialized for whatever task is desired. Prompt engineering is the process of creating ideal prompts for a desired output. Context, when talking about LLMs, is the information given to an LLM in a prompt that helps it generate better responses. Giving good context is a part of good prompt engineering.

The game demo for testing LLM dialogue is made in the Godot Game Engine using C# for scripting.

3 Prior Work

This section will review recent academic literature which relates to LLMs in video games. This literature review does not follow a systematic approach, however, the process will

be described here. The articles discussed were found using the following search query in multiple online databases: "("Video games" OR "games") AND ("LLM" OR "Large Language Model")". The articles were then selected manually, and any articles that did not discuss real-time dialogue generation were disregarded.

3.1 Use Cases

This section will explain the common use cases for LLMs in video games discussed in the academic literature. These include real-time dialogue generation, narrative generation, and procedural game world generation.

3.1.1 Dialogue Generation

This topic is the most relevant for this paper since this project will focus on creating dialogue with LLMs. There are two options for LLM generated dialogue. The LLM can generate dialogue during the game development process which would then be implemented into a fixed dialogue system. The other option is to use the LLM for real-time dialogue that is generated while the game is played. The second option is much more revolutionary than the first as it allows for a completely new way to control game dialogue. Because of this, most recent research in this field examines the real-time generation of LLM dialogue[9][1][6][5].

3.1.2 Narrative Generation

Narrative generation in video games using large language models focuses on the dynamic creation of plotlines, quests, and story arcs rather than just dialogue. By leveraging their ability to understand and produce coherent long-form text, LLMs can generate entire narrative structures that adapt to player actions and in-game events. This enables a shift away from rigid, pre-scripted storylines toward more fluid, emergent storytelling experiences. Games can present players with procedurally generated plots that still maintain thematic consistency, pacing, and character development. LLM-driven narrative generation holds the potential to greatly enhance player agency and replayability, but it also introduces challenges in ensuring narrative coherence, balancing player freedom with plot structure, and integrating generated content with game mechanics and world-building.

3.1.3 Procedural Generation

Procedural generation in video games using large language models enables the creation of rich, varied game content such as quests, lore, item descriptions, and environmental storytelling elements in real time. Unlike traditional procedural systems that rely on handcrafted rules and templates, LLMs can generate content with greater linguistic diversity and contextual awareness, resulting in more immersive and believable worlds. For example, an LLM can create a unique backstory for a village, generate side quests that reference recent player actions, or describe ancient ruins with tone and style consistent with the game's setting. This allows for high variability without sacrificing narrative depth. However, challenges include maintaining internal consistency, avoiding repetition or contradictions, and ensuring that generated content aligns with game design goals and player expectations.

3.2 Types of Games

This section will evaluate the types of games that previous articles have written about. Some common themes among these papers are the focus on role-playing games and virtual/extended reality.

3.2.1 Role-Playing Games

One of the most frequently mentioned genres of video games in video game LLM papers are role-playing games (RPGs). There is some debate over the definition of role-playing game, like whether the player-character must be a blank slate, but this paper will use the following definition. Role-playing games are games in which the player chooses player attributes such as skills, stats, appearance, and personality. The player's attributes and choices will affect the gameplay and narrative of the game. This genre of video game would benefit massively from the successful integration of LLMs since much of the gameplay of RPGs is dialogue. CRPGs (Computer RPGs) originate from TTRPGs (Table-top RPGs) like Dungeons and Dragons in which the Dungeon Master acts as the NPCs. This allows for incredible emergent gameplay and dynamic NPC-player interaction, since the NPC dialogue is improvised and acted on by a person. CRPGs lack this dynamic and unpredictable NPC interaction. LLMs could potentially add an AI dungeon master to CRPGS and simulate the randomness and unpredictability of real conversation. CRPGs mostly use scripted dialogue which is less interactive than real conversation, since there are only a few dialogue options for the player to choose and a few responses that the NPC can give. Comparing this with an LLM operated NPC where the player can say anything and the NPC can give real-time responses. Clearly an LLM NPC would be much more interactive in a game and simulate real life conversation more accurately. This would make the game world more believable and give the player the feeling of having more control over their character and their conversations with NPCs.

3.2.2 Extended Reality

Extended reality is an umbrella term which covers virtual reality, augmented reality, and mixed reality. It is a topic which comes up frequently in the academic research of LLMs in games and for good reason. The goal of implementing LLM-NPCs is to create a more immersive and lifelike world[4]. This complements extended reality as it too seeks to increase player immersion in the game/media world. Many studies use text-to-speech and speech-to-text for their dialogue systems. This simulates reality better than typing dialogue into a textbox or choosing dialogue from dialogue options. Text-to-speech offers the ability for LLM dialogue to be spoken by an AI voice, so that the NPC can talk to the player. This paper suggests that using text-to-speech actually harms player immersion since AI-voices often fall into the uncanny valley and players can immediately identify an AI generated voice line. Additionally, many studies have identified the delay of LLM response generation to be a technical problem with LLM integration in games. This causes a problem in extended reality games since players are expecting NPCs to act more life-like. In non-extended reality games, say for example a game with text boxes for dialogue, the delay is not as much of a problem since the player will not be expecting lifelike speech behavior from NPCs. The problem of LLM response time can also be mitigated with rolling text, so that the response is given to the player as the LLM is generating it, much like Chat-GPT 4 does.

4 Methods

This section lays out the approaches that will be taken in constructing the project.

The first question that this paper aims to answer is: "Can LLMs generate coherent and narratively consistent dialogue for video games?" Coherent dialogue means that what the NPC says makes sense in the context of the conversation. For instance, if the player asks an NPC a question, the NPC will give a response in-character. Narratively consistent dialogue means that the NPC dialogue makes sense in the context of the overall narrative of the game. For example, if the player asks an NPC to give them information about something, that NPC should only give information that the NPC should know.

4.1 Selecting a Model

In order to increase coherency and narrative consistency, several methods will be used. Firstly, choosing a large language model that specializes in conversational responses is important. Several models will be examined and tested. For each model, several categories of competence will be tested: contextual accuracy, literary quality, and speed. The model which scores highest in these categories on average will be selected for use in the game demo.

4.1.1 Contextual Accuracy

To test contextual accuracy, 100 responses will be evaluated for errors. Errors include reciting incorrect information about the NPC or the game world, reciting information that is outside the scope of the context window, and giving responses that do not align with the NPC's personality traits or relationship with the player. Each LLM will be given a score out of 100.

4.1.2 Literary Quality

Literary quality is harder to test since it is subjective. What one person deems good quality literature may not be the same for all players. User testing will be performed to find what the majority considers to be good quality. At least 10 testers will examine 100 LLM responses for each model and give each response a score of 1 to 5 for quality of writing. Users will be asked not to factor in context or coherence into their scores. The average score will be calculated for each model.

4.1.3 Speed

Speed will be determined by a custom testing program. The average response time will be calculated for at least 100 responses for each model.

4.2 Customization of the Model

Once a model has been selected, the model must be customized to make it specialized for dialogue generation. Several steps will be taken to achieve this. First, the model will be fine-tuned on a smaller curated dataset. Then, a retrieval-augmented wrapper will be implemented to make responses more world accurate for static world lore. For dynamic narrative memory, a memory management system will be implemented. Finally, a curation program will be written to catch any responses that are problematic and query the LLM again for another response.

4.2.1 Fine-Tuning

The purpose of fine-tuning for this project is to have the model speak in character, respond in short quick responses, and stay within the genre. I will first create a dataset for the model either by writing example dialogues myself or by training it on existing data like books, video game transcripts, etc. These datasets could also be used together. This project will use LoRA or QLoRA fine-tuning since it can

be done with less resources than fully fine-tuning which takes multiple GPUs and more time. Low-Rank Adaptation (LoRA) fine-tuning only trains small adaptive layers over the base model. This means that the base model retains its language processing skills and you can switch between many different personalities, which is important for generating interesting dialogue for multiple NPCs. After each fine-tuning session, the model will be tested and put through an iterative process. Once a certain threshold for contextual accuracy, literary quality and speed is achieved or no further improvement is made in multiple training sessions, the fine-tuning process will be completed.

4.2.2 Retrieval Augmented Memory

Retrieval Augmented Memory or RAG is a technique for improving LLM accuracy for specific static datasets, like company information. For this project, RAG will be used to help the model recall information about the world's lore like place-names, item descriptions, and well-known characters as discussed by Kostilainen (2024) [10]. Everything that all characters should know and that will not change throughout the game can be placed into a dataset that will be used for RAG. The model will be iteratively tested for accuracy of information regarding the custom RAG datasets. RAG can cause extra latency for generation since it adds extra overhead to find information from an external database. This latency will be recorded and tested later for effects on user experience.

4.2.3 Memory Management System

A memory management system, MMS from now on, is similar to RAG in that it relies on external data to be searched after prompting and before generation. An MMS allows for long-term dynamic memory storage. For instance, previous dialogue with a certain NPC can be stored and used for new dialogue generation. In addition, plot points and narrative information can be stored in an MMS. Importantly, an MMS is continually updated as the game goes on, whereas RAG datasets are static.

4.2.4 Response Curation

The final step in customizing the model will be to implement a response curation program. This program will filter problematic dialogue. It will flag responses that contain certain keywords. For example, if a certain NPC is not supposed to know about another character, they should not say their name in dialogue unless a player gives them the name. In addition, inappropriate responses will be filtered with keywords. When a response is flagged the system will automatically send another query to the LLM stating why the response was flagged and requesting another response to the player's dialogue.

5 Evaluation Metrics

The goal of this project is to generate real-time, coherent, and narratively driven dialogue. In order to evaluate the success of the project, these three criteria will be tested individually. The testing will follow the same steps as the testing for selecting a base model. Contextual accuracy will be determined by analyzing 100 dialogue responses for contextual specific questions and categorizing them into contextually accurate and not contextually accurate responses. A percentage will be calculated to show the contextual accuracy of the LLM dialogue. Next, the literary quality will be evaluated through a process of user testing. A group of at least 10 evaluators will give ratings of 1-5 for each dialogue response regarding the literary quality of the LLM dialogue. All ratings will be averaged to give a score out of 5 for literary quality. The speed of the LLM will be assessed with a program that averages the dialogue latency for 1000 dialogue responses.

Once the dialogue has been assessed with these metrics, there will be user testing for overall experience. In a survey, players will rate their experience with the dialogue interface, the integration of the gameplay and dialogue systems, the narrative quality regarding dialogue, and the overall experience of interacting with LLM NPCs in game. At least 10 playtesters will be surveyed.

6 Ethical Considerations

This section will discuss the ethical implications of using Large Language Models to create video game dialogue.

6.1 Bias

One major issue with LLMs is the unpredictable nature of their responses. They generate response from massive datasets that are difficult to curate. As a result, some problematic data can be used in the training process and cause unwanted effects such as inappropriate or controversial responses. Similarly, bias can be unknowingly trained into an LLM if the data is poisoned[14]. In 2023, the World Association for Artificial Consciousness conducted several studies to locate bias in the most popular LLMs. They tested mainstream models for regional, racial, and age bias using evaluation tools and datasets. These studies conclude that the top LLMs have a low to moderate level of regional bias[8], a low to moderate level of racial bias[7], and a moderate to high level of age bias[17]. These biases can exacerbate social inequalities when LLMs are used in the wrong situations. For example, job application screenings are being

done more and more by LLMs which may have biases. This could cause unintentional discrimination against certain applicants. There are many more applications of LLMs which could be problematic, considering potential biases, like insurance, judicial and legal assistants, customer chat bots, educational chat bots, etc.

6.2 Security and Privacy

There are many security and privacy risks involved with the use of large language models. This paper will not discuss every security threat to LLMs generally, but will address issues specific to LLM generated video game dialogue. The specific threats identified in this paper are inference attacks, prompt injection, and extraction attacks. Inference attacks are attacks that seek to obtain information about LLM training data by inference[3]. These attacks do not seek to gain training data directly, but use inference to gain secondary information. This type of attack is a concern for this project since an adversary who successfully carries out this attack could gain information about the fine-tuned training datasets and exploit this information for gain within the game. A survey of papers discussing LLM vulnerabilities found that fine-tuning adaptive layers rather than the head of the model reduces the risks of membership inference attacks[16]. This is not much of an issue in single-player games, but for a competitive multiplayer game, the adversary might gain an advantage over other players, creating an unfair gaming experience. A similar issue arises with the threat of prompt injection, which occurs when an adversary provides input to the LLM that generates harmful or unintended responses, such as leakage of sensitive information about other players or about the game[12].

6.3 Content Moderation

Content moderation is an essential step in ensuring a safe gaming environment for players. LLM dialogue is generated dynamically, increasing the risk of harmful, offensive, and inappropriate responses. To mitigate this risk, a combination of careful prompt engineering and rule-based moderation is required. There has been some exploration of the use of LLMs as content moderators themselves. One study found that the median precision was 64% and the median precision was 83% for several popular LLMs regarding the moderation of content in online forums[11]. This is better than nothing, but for full protection against harmful content, it is advisable to implement a rule-based system. It is also important to detect and respond to edge cases such as prompt injection, especially in multiplayer games.

6.4 Creative Authenticity

There is much discussion about the integrity of LLM generated content. The issue extends to all fields in which LLM are being used, especially in creative fields. Some people argue that LLMs are inauthentic and simply regurgitate what other people have written. This issue is especially significant in academia[15][13]. While large language models are derivative, meaning they generate responses based on previous data, they offer writers the opportunity to generate massive amounts of essentially filler material. When used in conjunction with human written narrative and storytelling, they can be used to fill in a world so that it feels more real while maintaining the narrative of the human writers.

6.5 Player Manipulation

Player manipulation is an ongoing issue in video game design. Certain features of games are designed to manipulate players into continuing to play the game, watching advertisements, and spending money. This can be malicious if the methods become predatory or unfair to the player. One issue that is particularly egregious, yet has not received much legal attention is video game gambling. Many games have adopted a loot-crate monetization model in which players spend real world money for a randomized digital content reward. The House of Lords in the UK states that they do not consider this gambling since there is no option to cash out for real money[2]. However, for many games there are third party websites which are used to buy and sell accounts for real money, offering players the ability to cash out. Whether or not there is real money involved, the psychological effects of these systems are something to be considered. This issue is pertinent to LLMs in games since LLMs can be subtly used to persuade players to act differently. Players may be convinced to buy an in game item or play for a bit longer. This use of LLMs can quickly become malicious and harm the user's experience.

7 Timeline

Here is the proposed timeline for this project:

- June
  - Work on game design document
- July
  - Finish game design document

- August

  – Week 1-2: Choose model, fine-tune
  – Week 3-4: Work on retrieval augmented generation system and memory management system

- September

  – Week 1-2: Continue working on RAG and MMS
  – Week 3-4: Begin user testing

- October

  – Week 1-2: Continue user testing and working on RAG and MMS
  – Week 3-4: Finish RAG and MMS

- November

  – Week 1-2: Begin working on filtering system
  – Week 3-4: Finish filtering system perform final user testing

- December

  – Week 1-2: Work on poster and final polishing of game demo
  – Week 3-4: Final submission

References

[1] Akoury, Nader, Yang, Qian, and Iyyer, Mohit. "A Framework for Exploring Player Perceptions of LLM-Generated Dialogue in Commercial Video Games". In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Bouamor, Houda, Pino, Juan, and Bali, Kalika. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 2295–2311. DOI: 10.18653/v1/2023.findings-emnlp.151. URL: https://aclanthology.org/2023.findings-emnlp.151/ (visited on 03/01/2025).

[2] *Are Loot Boxes Gambling and Therefore Addictive?* en-US. Jan. 2024. URL: https://kindbridge.com/gaming/are-loot-boxes-gambling-and-therefore-addictive/ (visited on 04/29/2025).

[3] Birch, Lewis et al. *Model Leeching: An Extraction Attack Targeting LLMs*. arXiv:2309.10544 [cs]. Sept. 2023. DOI: 10.48550/arXiv.2309.10544. URL: http://arxiv.org/abs/2309.10544 (visited on 04/28/2025).

[4] Christiansen, Frederik Roland et al. "Exploring Presence in Interactions with LLM-Driven NPCs: A Comparative Study of Speech Recognition and Dialogue Options". In: *Proceedings of the 30th ACM Symposium on Virtual Reality Software and Technology*. VRST '24. New York, NY, USA: Association for Computing Machinery, Oct. 2024, pp. 1–11. ISBN: 979-8-4007-0535-9. DOI: 10.1145/3641825.3687716. URL: https://dl.acm.org/doi/10.1145/3641825.3687716 (visited on 02/28/2025).

[5] Cox, Samuel Rhys and Ooi, Wei Tsang. "Conversational Interactions with NPCs in LLM-Driven Gaming: Guidelines from a Content Analysis of Player Feedback". en. In: *Chatbot Research and Design*. Ed. by Følstad, Asbjørn et al. Cham: Springer Nature Switzerland, 2024, pp. 167–184. ISBN: 978-3-031-54975-5. DOI: 10.1007/978-3-031-54975-5_10.

[6] Csepregi, Lajos Matyas. "The Effect of Context-aware LLM-based NPC Conversations on Player Engagement in Role-playing Video Games". en. In: *FIXME* (FIXME).

[7] Duan, Yucong. "Large Language Model (LLM) Racial Bias Evaluation". en. In: ().

[8] Duan, Yucong et al. ""Ranking of Large Language Model (LLM) Regional Bias" – DIKWP Research Group International Standard Evaluation". en. In: ().

[9] Huang, Junyang. "Generating dynamic and life-like NPC dialogs in role-playing games using large language model". eng. In: *FIXME* (2024). Accepted: 2024-06-12T08:14:59Z. URL: https://lutpub.lut.fi/handle/10024/167809 (visited on 03/01/2025).

[10] Kostilainen, Sami. *Next generation of NPC dialogue: creating responsive NPCs (Non-Player Characters) with Retrieval-Augmented Generation and real-time player data*. eng. fi=Ylempi AMK-opinnäytetyö—sv=Högre YH-examensarbete—en=Master's thesis—. Accepted: 2024-06-03T08:48:46Z. 2024. URL: http://www.theseus.fi/handle/10024/861943 (visited on 03/01/2025).

[11] Kumar, Deepak, AbuHashem, Yousef Anees, and Durumeric, Zakir. "Watch Your Language: Investigating Content Moderation with Large Language Models". en. In: *Proceedings of the International AAAI Conference on Web and Social Media* 18 (May 2024), pp. 865–878. ISSN: 2334-0770. DOI: 10.

1609/icwsm.v18i1.31358. URL: https://ojs.aaai.org/index.php/ICWSM/article/view/31358 (visited on 04/29/2025).

[12] Liu, Yi et al. *Prompt Injection attack against LLM-integrated Applications*. arXiv:2306.05499 [cs]. Mar. 2024. DOI: 10.48550/arXiv.2306.05499. URL: http://arxiv.org/abs/2306.05499 (visited on 04/29/2025).

[13] Meça, Alba and Shkëlzeni, Nirvana. "Academic Integrity in the Face of Generative Language Models". en. In: *Emerging Technologies in Computing*. Ed. by Miraz, Mahdi H. et al. Cham: Springer Nature Switzerland, 2024, pp. 58–70. ISBN: 978-3-031-50215-6. DOI: 10.1007/978-3-031-50215-6_5.

[14] Pathmanathan, Pankayaraj et al. *Is poisoning a real threat to LLM alignment? Maybe more so than you think*. arXiv:2406.12091 [cs]. Feb. 2025. DOI: 10.48550/arXiv.2406.12091. URL: http://arxiv.org/abs/2406.12091 (visited on 04/29/2025).

[15] Perkins, Mike. "Academic integrity considerations of AI large language models in the post-pandemic era: ChatGPT and beyond". In: *Journal of University Teaching and Learning Practice* 20.2 (Feb. 2023). Publisher: Open Access Publishing Association (OAPA), pp. 1–24. DOI: 10.3316/informit.T202411130000959175171095. URL: https://search.informit.org/doi/abs/10.3316/informit.T202411130000959175171095 (visited on 04/29/2025).

[16] Yao, Yifan et al. "A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly". In: *High-Confidence Computing* 4.2 (June 2024), p. 100211. ISSN: 2667-2952. DOI: 10.1016/j.hcc.2024.100211. URL: https://www.sciencedirect.com/science/article/pii/S266729522400014X (visited on 04/28/2025).

[17] Yucong Duan et al. ""The Large Language Model (LLM) Bias Evaluation (Age Bias)" –DIKWP Research Group International Standard Evaluation". en. In: (2024). Publisher: Unpublished. DOI: 10.13140/RG.2.2.26397.12006. URL: https://rgdoi.net/10.13140/RG.2.2.26397.12006 (visited on 04/27/2025).