Development of an LLM-first 2D Video Game

Using Language Models to Create Dynamic Game Worlds

Graduate

Elena Gerber

Noël Joost

Initial Situation: Traditional role-playing video games (RPGs), where players assume the role of a character in a fictional world, often rely on static dialogues and storylines. This can make replaying the game feel repetitive and less engaging, as the story and conversations remain the same. In contrast, tabletop RPGs thrive on the creativity of the human game master, who continuously invents new settings and stories and allows players to interact naturally with the world. With the rise of large language models (LLMs), new opportunities are emerging for digital games. A key strength of modern LLMs is their ability to generate human-like text in response to natural language input. Integrating LLMs into games enables live, context-sensitive responses to player actions and dialogue, bringing some of the freedom and spontaneity of tabletop RPGs into video games. This project aims to develop a 2D RPG with a top-down view that combines traditional gameplay with the creative power of LLMs. At the start of each game, an LLM generates a unique world. During gameplay, characters respond to the player in real time, meaning no playthrough is the same. For developers, this moves focus away from scripting fixed content to designing systems and prompts that enable the AI to take over parts of the storytelling.

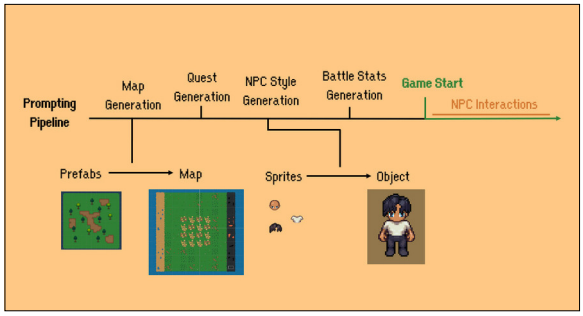
Approach / Technology: The game is developed in Unity using a 2D pixel art style and includes typical RPG elements such as story, exploration, non-playable characters (NPC), battles, items and quests. Game content is partly generated using LLMs and partly built from predefined assets. For example, character parts like heads, bodies and clothes are defined in advance, and the LLM selects from these to create NPCs. Similarly, the map is built from predefined 30x30 tiles depicting elements like forests, plains and villages, which the LLM assembles into a complete world. The generation process starts with a prompt defining the map, followed by prompts for quests, NPCs and items. NPC appearance and battle skills are generated in separate prompts to keep outputs manageable. For simpler tasks like NPC visuals, skills and interactions, we use OpenAI's cost-efficient GPT-4o-mini. For tasks requiring more consistency and depth, such as map and quest generation, we use the o3-mini model.

Result: We built a functional top-down 2D RPG with LLM-supported elements that generate new content on each playthrough. Our key finding is that the quality of generated content depends heavily on the context provided. Without sufficient details, quests and characters often become repetitive or inconsistent. For example, NPCs might block quest progression or items can appear in incorrect locations. While these issues occasionally caused confusion, the eight people we selected to test our game generally responded positively, appreciating the dynamic nature of the experience. A major challenge we encountered was balancing the amount and complexity of prompts with system performance. Longer or more detailed prompts improve output coherence but increase the risk of missing details. Splitting input into smaller prompts helps but raises processing time. A promising approach is sending multiple prompts simultaneously for parallel processing, which could improve content accuracy and reduce wait times. Overall, this project indicates that a thoughtful combination of AI and traditional techniques can open exciting new possibilities for creating more dynamic and imaginative games in the future.

Generated dialogue with an NPC.
Own presentment

Battle with generated NPC and skills.
Own presentment

Diagram illustrating how the LLM is used throughout the game.
Own presentment



Advisors
Prof. Dr. Mitra Purandare, Benjamin Plattner

Co-Examiner
Dr. Marc Röösli, Mettler-Toledo International Inc., Greifensee, ZH

Subject Area
Artificial Intelligence