AI DUNGEON STORY GENERATOR – PROJECT REPORT

Abstract

This project presents an AI-powered interactive story generator leveraging the GPT-Neo language model and Streamlit framework. It allows users to create immersive, multi-genre narratives by providing a prompt and genre selection. The generator produces multiple story continuations and offers downloadable text files. This project demonstrates the power of modern transformer models in creative writing and user engagement through a sleek web app.

Introduction

The art of storytelling has evolved from fireside tales to digital narratives crafted by artificial intelligence. The AI Dungeon Story Generator is designed to harness the cutting-edge GPT-Neo model to breathe life into user-generated story ideas. By combining intuitive user inputs with AI creativity, the tool democratizes narrative creation, allowing anyone to become a storyteller in genres ranging from fantasy to horror. This project aims to blend human imagination with machine intelligence to unlock new storytelling horizons.

Tools Used

- Python: The backbone of the project, orchestrating model loading, input processing, and output generation.
- Transformers Library (Hugging Face): Provides access to GPT-Neo (1.3B), a powerful text-generation transformer model.
- Streamlit: A lightweight web app framework enabling rapid UI development and seamless interaction with AI models.
- NLTK: Used for text processing and tokenization during development.
- Other: Standard libraries such as random, datetime, and io for auxiliary tasks like randomization, timestamping, and file handling.

Steps Involved in Building the Project

1. Model Selection and Loading: GPT-Neo 1.3B was chosen for its open-source accessibility and strong language generation capabilities. The model is loaded efficiently using Streamlit's caching to reduce wait times.

2. User Interface Design: Streamlit forms and widgets are used to create an intuitive input interface for users to select genres and enter story prompts. A sidebar offers instructions to enhance usability.

3. Prompt Engineering: Genre-specific opening lines set the thematic tone. The user's input is appended to these prompts to guide the model in generating contextually relevant stories.

4. Story Generation: Using the Transformers pipeline, multiple story continuations are generated with controlled randomness (temperature, top-p, top-k sampling) to maintain creativity while avoiding repetition.

5. Post-Processing: Generated stories are checked for endings; if absent, a poetic closing statement is appended to give a satisfying narrative conclusion.

6. Display and Download: Generated stories are styled and displayed within the app using HTML/CSS for readability. Users can download the full set of stories as a timestamped text file.

Conclusion

The AI Dungeon Story Generator successfully bridges the gap between AI and human creativity, enabling interactive and genre-flexible storytelling. By leveraging the power of GPT-Neo and Streamlit, the project demonstrates how accessible AI can transform narrative creation. Future improvements could include expanding genre options, integrating user feedback loops, and enhancing story coherence with advanced prompt engineering or fine-tuning.