NAME: Ahmed hahsem

student number: 210208945

Generative AI for Story Generation

1. Introduction

This project is about using Generative AI methods which mainly refers to retraining a Large Language Model (LLM) that was originally trained for creative text generation. The aim was to create a model that would be able to provide coherent, long texts when a user provided a prompt. We have settled on using GPT-2, which is a language-model that runs on transfomers and was built by the people over at OpenAI, because it has the best combination of both power and computational efficiency.

2. Objective

The goal of the project was to take the GPT-2 model and adjust it to the greatest extent possible using multiple long story datasets and to create a capable generator that can generate a new user message based on a prompt. The model would have to be able to cleverly and efficiently figure the new sub-text of a certain topic out and then write the text in various known genres such as sci-fi, horror, fantasy, and adventure.

3. Dataset

The creation of a unique dataset entitled long_stories_40k.txt was a byproduct of compiling 40,000 lines of comprehensible stories. The data was loaded onto Google Colab and then processed line by line following the removal of white space and the filtering out of empty lines.

- Size: 40,000 story samples
- Format: Plain text
- Use Case: Fine-tuning GPT-2 on narrative generation tasks

4. Methodology and Tools

We performed the project's task using Python and Hugging Face Transformers library. The course of action that we underwent was:

- Installed the required packages (transformers, datasets)

- Employed the Dataset class from Hugging Face to load the dataset
- Used GPT2Tokenizer to tokenize the data with truncation and maximum length of 128
- Assigned the padding token to the EOS token
- Downloaded the standard GPT-2 model (GPT2LMHeadModel)
- Employed DataCollatorForLanguageModeling with masked language modeling off (mlm=False)
- Set up the TrainingArguments with:
- 2 epochs
- Batch size = 4

5. Training and Results

The model was run on a Google Colab T4 GPU for 2 complete epochs. The loss throughout the training decreased progressively from around 1.49 to 0.15, indicating obvious learning with time.

Final training metrics:

- Steps: 20,000
- Final Loss: ~0.15
- Training Time: ~41 minutes
- Training Speed: ~32 samples

6. Output and Generation

The model saved after the training was reloaded to test it. Here is a prompt-based generation system that was created:

- The user gives a prompt (e.g., “magic castle”)
- The system automatically detects a category ([FANTASY], [SCI-FI], etc.)
- The prompt is added to the tag and sent to the model
- Then the model outputs a new story by first using top_k, top_p, and temperature sampling.

Example Output:

Prompt: magic

Generated:

[FANTASY] magic cursed knight sought redemption by questing for a legendary sword said to break all enchantments and restore his lost honor. Danger lurked in every shadow, but they pressed onward.

7. Challenges Faced

The project was faced with various technical difficulties:

- Colab Crashes: The Colab runtime was switched off several times while training.
- GPU Timeout: During some runs, training was terminated because of inactivity or timeouts.
- Memory Limitations: Optimization of memory was needed to manage a dataset of 40k examples.
- Progress Reset: Loss of progress happened if the browser session got closed.

We coped with these by mostly re-installing checkpoints, creating model files, and modifying the batch size within memory capacity.

8. Conclusion

This project showed how a model, the LLM (GPT-2) fine-tuned transformer-based, can be used to generate stories in specific themes. The latest model was able to gain experience from a user's prompt and generate various and high-quality results in quick time in different domains of writing. The authors managed to solve all the technical issues that appeared during the encoding of the underlying data and were quite successful in reaching the goal of the project, so that it could be a step for the development of a web app or integration into chatbot.