IS640 GPT Project

Class GPT 1 Project

# Summary

Build a similar GPT 1 architecture using Andrej Karpathy's [YouTube](https://www.youtube.com/watch?v=kCc8FmEb1nY).

As you follow the video, you will go through milestones that use different practices to improve the model.

# Project Overview

Students will work in groups to develop and fine-tune a basic LLM on a dataset of their choosing. They will train the model to generate coherent text aiming to balance the quality of the output with computational efficiency through each iteration. The project involves understanding model architecture, training techniques, evaluation, and improving the model's performance iteratively.

## Key Learning Goals

1. Understand Transformer Architecture
   * Learn about self-attention, embeddings, positional encodings, and the overall LLM architecture.
2. Dataset Preparation
   * Learn how to clean, pre-process, and tokenize text datasets.
3. Model Training
4. Text Generation and Evaluation
5. Iterative Improvements
   * Experiment with fine-tuning parameters.
   * Consider the trade-offs between model performance and computational efficiency.

## Criteria

1. Create an iteration of each milestone and commit as a group.
2. Each group member must make a commit.

# Milestones

## Milestone 1: Dataset Exploration and Preparation

* Select a dataset (e.g. movie scripts, Reddit comments, books).
* Clean and tokenize the data.
* Justify dataset choices in a project report.

## Milestone 2: Basic Model Usage (Bigram Language Model)

* Use the provided Bigram Language Model to start.
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone2.txt

## Milestone 3: Self-attention & SoftMax Iteration

* Update the provided model to include Self-attention Iteration
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone3.txt

## Milestone 4: Multi-head Attention

* Update the provided model to include Multi-head Attention Iteration
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone4.txt

## Milestone 5: Feed Forward Layers

* Update the provided model to include Feed Forward Layers
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone5.txt

## Milestone 6: Residual Connections

* Update the provided model to include Residual Connections
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone6.txt

## Milestone 7: Layer Normalization

* Update the provided model to include Layer Normalization
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone7.txt

## Milestone 8: Dropout

* Update the provided model to include Dropout
* Train the model on the selected dataset.
* Track performance metrics such as loss during training.
* Print out average training loss and validation loss at the end
* Print out generated tokens/text to preview the current state.
* Save generated text of 300 - 500 tokens to a file called milestone8.txt

## Final Presentation

* Present the model’s performance, the generated text, and the challenges faced.
* Change parameters and see if you can get better improvements on loss and generated text.