Project Report

SastaGPT - A Simplified Transformer-Based Language Model

In this project, I developed an autoregressive character-level language model named SastaGPT, a very simple version of an LLM, capable of generating text that mimics the style of the input dataset. Initially, I started with simple bigram models that just take into account the very previous character but that gave bad results even with low loss values. It was evident that we could do better.

What the bigram model really lacked was the concept of attention. Think of it as the affinity between tokens(in our case just characters). We first establish the connection between tokens and then move to the calculation and optimization part of the neural network. Unlike simple bigram models, which struggle to produce coherent and structured text, SastaGPT leverages multi-headed self-attention to generate meaningful and contextually relevant sequences. I trained my model on the complete works of Shakespeare and classical Stoic texts. What it returned as output surprised me. Although being a 10 Million parameter (compared to Billions parameter level models), the resulting text had a really good similarity with the works it was trained on.

Note: I already discussed using libraries like PyTorch or Tensorflow for my project with Professor. Since I was experimenting with advanced architectures like RNN and Transformers, Professor gave me the permission to use PyTorch for my project. It would not be feasible to make this project without using PyTorch.

Literature Review

The project draws inspiration from “Attention Is All You Need” by Vaswani et al., which introduced the Transformer architecture. The Transformer’s self-attention mechanism enables it to capture long-range dependencies in sequences, making it superior to traditional RNN-based models.

Dataset Source and Description

For training, I curated two primary datasets:

- Shakespeare’s Works: Downloaded from the Internet Archive, containing all of his plays and poems combined into a single text file.

- Stoic Philosophy Texts: Included Meditations by Marcus Aurelius, Letters from a Stoic by Seneca, and Enchiridion and Discourses by Epictetus. These texts were already processed, requiring minimal cleaning. So I also clubbed them in another text file.

Data Exploration and Important Features

Since this is a character-level model, the data was processed into sequences of characters. Important aspects included:

- Character Frequency Distribution: Understanding which characters appeared most frequently helped shape the vocabulary.
- Contextual Dependencies: The length of context used (window size) played a crucial role in generating coherent text.

Methods

My final model uses a transformer to predict the next character in an auto regressive manner. Transformers are a deep learning model architecture that relies on self-attention mechanisms to process sequential data. Unlike RNNs, which process data sequentially, Transformers allow for parallel processing of inputs, making training significantly faster.

Multi-Headed Self-Attention

Imagine reading a book with multiple highlighters, each highlighting different aspects of a sentence—one for important words, another for grammatical structure, and yet another for emotional tone. This is similar to multi-headed self-attention, where multiple attention heads focus on different relationships within a sequence.

Implementation Details

- Bigram Model: Initially, I trained a simple bigram model where each character predicts the next one based on frequency counts.
- Transformer-Based Model (SastaGPT): Implemented using multi-headed self-attention, allowing the model to understand contextual dependencies beyond immediate neighbors.

Hyperparameter Tuning

I experimented with different hyperparameter settings to get the most optimal result.

- Optimal settings: BATCH_SIZE = 64, CONTEXT_WINDOW = 256, but these were computationally expensive. It took me 2 hours to complete just 500/5000 epochs with these settings and my M1 machine could not handle it. So I had to significantly reduce these parameters to get some results.

- Final settings:
    - BATCH_SIZE = 32
    - CONTEXT_WINDOW = 8
    - EPOCHS = 5000
    - LR = 3e-4
    - EMBEDDING_DIM = 384
    - HEADS = 6
    - LAYERS = 6
    - DROPOUT_RATE = 0.2

Due to hardware constraints, I reduced the batch size and context window to ensure training feasibility. The surprising part still is the quality of results that came out even though it was trained with such constraints and on a tiny dataset. Instead of looking like pure gibberish, the output files look very similar to English compositions.

Final Results

The trained model successfully generated text that resembled the input datasets.

- Shakespearean-style output: The model produced text that followed archaic linguistic patterns and poetic structures.
- Stoic-style output: Generated philosophical reflections that mimicked the tone of Stoic teachings.

While the model does not generate perfectly correct English, the output words resemble natural language patterns rather than random character sequences.

Conclusion

SastaGPT demonstrates the power of Transformers in a distilled form. By leveraging self-attention, the model achieves a level of coherence far beyond traditional n-gram models. While performance was limited by hardware constraints, future work could involve cross attention, optimizing memory usage or training on more powerful GPUs.

References

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). "Attention Is All You Need."
- Andrej Karpathy Neural Networks Series
- Internet Archive: Shakespeare’s Works and Stoic Philosophy Texts

GitHub Repo:

Please find the respective GitHub Repo here: https://github.com/alwaysafoujdar/SastaGPT