Poem generation using GPT-2

Augustin Cablant
ENSAE Paris
augustin.cablant@ensae.fr
Poetry Analysis and Generation Notebook

Abstract

This project explores the generation of poems using a transformer-based model (GPT-2), with a special focus on incorporating poem categories (e.g., sonnets, haikus) and author information into the model inputs. We present a custom dataset of poetry, a fine-tuned GPT-2 Lo et al. [2022] model with reduced parameters, and analyze the performance of the generation both qualitatively and through model optimization techniques such as quantization.

1 Introduction

Natural language generation has seen remarkable progress with the advent of transformer-based models such as GPT-2 Lo et al. [2022]. While general text generation has been widely studied, poetry generation remains a niche yet challenging sub-field due to the stylistic, rhythmic, and thematic constraints inherent in poetic forms Lau et al. [2018]. This project aims to create a controllable poem generation model capable of producing coherent verses conditioned on hierarchical metadata such as author and category.

2 Related Work

Poetry generation has been addressed using various approaches, ranging from rule-based systems to RNNs and transformers. Recent models leverage pretrained language models for creative text generation. For instance, Lau et al. [2018] explore style-aware neural generation, while **?** focus on neural poetry generation with rhyme and meter constraints. Our work builds upon GPT-2 and adapts it for poetry by enriching inputs with structured metadata.

3 Data and Preprocessing

The corpus comes from a dataset publicly available on Kaggle[1], and consists of annotated poems in txt format, each including the poem text along with metadata such as author, main category, and sub-category. We preprocessed the text by:

- Concatenating metadata with the poem text using a structured format.
- Removing excessive newlines and punctuation.
- Tokenizing and padding using GPT-2's tokenizer, setting a maximum sequence length of 400.

4 Model and Training

GPT-2 (Generative Pre-trained Transformer 2) is a transformer-based language model that generates text by predicting the next word in a sequence, given its previous context. It is based on the Transformer architecture, which uses self-attention mechanisms to process and generate sequences of text. The core idea behind GPT-2 is to use a large-scale pretraining task on a general corpus of text, followed by fine-tuning on specific tasks, such as poetry generation.

4.1 Transformer Architecture

The Transformer architecture consists of multiple layers of self-attention and feed-forward neural networks. Each layer computes a set of attention scores to determine how much focus each word in

[1] https://www.kaggle.com/datasets/michaelarman/poemsdataset/data

the sequence should receive from other words. The key equation for self-attention is:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$

Where: - Q is the query matrix, - K is the key matrix, - V is the value matrix, and - d_k is the dimension of the key vectors.

The self-attention mechanism allows the model to weigh different words' importance in a given context, even if they are far apart in the sequence.

4.2 Autoregressive Model

GPT-2 is an autoregressive model, meaning it generates text one token at a time, conditioning each token on the previous tokens in the sequence. The goal during training is to minimize the negative log-likelihood of the predicted word w_t given the previous context $w_1, w_2, \ldots, w_{t-1}$:

$$\mathcal{L} = -\sum_{t=1}^{T} \log P(w_t | w_1, w_2, \ldots, w_{t-1})$$

Where:

- w_t is the word at time step t,
- T is the total number of tokens in the sequence,
- $P(w_t | w_1, w_2, \ldots, w_{t-1})$ is the probability of the token w_t conditioned on the previous tokens.

The model is trained using a causal (unidirectional) attention mask, meaning that the prediction for token w_t can only depend on the tokens w_1 to w_{t-1}, and not on the future tokens $w_{t+1}, w_{t+2}, \ldots$.

4.3 Fine-Tuning on Poetry Generation

To adapt GPT-2 for poetry generation, we fine-tune the model on a dataset of poems. The model's parameters are initialized with the pretrained weights from the general language model and then updated using the poetry dataset. The dataset consists of poems annotated with metadata, such as author and category, which helps condition the generated text. The objective remains the same: minimize the cross-entropy loss:

$$\mathcal{L}_{\text{poetry}} = -\sum_{t=1}^{T} \log P(w_t | w_1, w_2, \ldots, w_{t-1}, \text{metadata})$$

Where the metadata (such as poem category or author) is encoded as additional input to the model, allowing it to generate poems with specific styles or themes.

4.4 Model Configuration

We use a reduced version of GPT-2 with the following configuration:

- 6 layers of Transformer blocks,
- 6 attention heads per layer,
- Embedding size of 384,
- Maximum sequence length of 400 tokens.

This smaller configuration allows for faster training while still capturing the essential structure of the language. The model is trained using the AdamW optimizer, with a learning rate of 2×10^{-5}, a batch size of 2, and a warm-up schedule for the learning rate.

4.5 Optimization and Deployment

For efficient deployment, we apply post-training dynamic quantization, where the model weights are converted to 8-bit integers (int8) to reduce the model's memory footprint Yao et al. [2022]. The model is also traced using TorchScript for faster inference on production hardware.

The training data was loaded using a custom PyTorch Dataset class that formats input with both textual and categorical metadata. Mixed precision training (AMP) was used to speed up training on GPU.

5 Results

We evaluate the model qualitatively. Generated samples respect the category and are often coherent. Below is a sample generated poem:

Prompt: "The wind whispers secrets"
Category: forms/haiku
Generated:
The wind whispers secrets,
No one night,
A moment never be to
And it makes the face.
It's no fear is so
Then they knew the door.
You must come back from the moon,
So that they are born.
My body goes down into rain
With your heart and sun

In our analysis of Shakespeare's sonnets, we generated a word cloud based on the frequency of words across the entire corpus. Unsurprisingly, the word "love" appears as one of the most frequent terms, reflecting the central theme of Shakespeare's poetry. Love is a prominent motif in his works, and this is especially evident in his sonnets, where emotions of love, longing, and loss are explored in depth. Given the recurrence of this word, we aim to generate new Shakespearean sonnets by conditioning the generation process on the first line of his existing sonnets, using this as a prompt. By doing so, we expect the generated text to maintain the thematic richness and stylistic qualities associated with Shakespeare's poetic exploration of love.

One of Shakespeare's most famous sonnets begins with the line "From fairest creatures we desire increase," and we attempted to generate the following lines based on this initial phrase.

Generation:
From fairest creatures we desire increase That thereby beauty's rose might never found?
I'll always come, but not even it be my life. I have no one and so can be true; the only me of them a soul and the world is there't see in every other way to have been to have to see.
I can not find myself to take all my mind so many love with your own hands as if the future of our hearts with this time that's only day. The last heart has got you feel at the man who does your eyes in the day from what that will leave to get out of all are beautiful life's name with a soul.
No, the Lord is on some of a new friend of those men, to know the words, nor a child of our life, is for me to me in each other've made their children, who was going, when my arms and love; and that I have the best, I am very dear, if I remember that I wish to be too. What they don're I've heard us do not have to show them for others or as one would help you to see about my soul again my days.

The result, however, deviated from Shakespeare's distinct style. We observed that the sentences were significantly longer, and the rhythm and structure didn't quite align with the traditional Shakespearean sonnet form. This discrepancy is likely due to the training corpus, which did not include a sufficient number of Shakespeare's own sonnets. Consequently, the model may not have fully captured the intricacies of Shakespeare's language and poetic style, leading to a less authentic generation.

6 Discussion and Limitations

While the model captures poetic form and category, it occasionally generates flat or repetitive lines. Rhyme and meter are not explicitly enforced. Future work could involve fine-tuning with rhyme constraints, evaluating via human annotation, or conditioning on emotional tone.

7 Conclusion

This project demonstrates the potential of category-aware poem generation using GPT-2. By conditioning on structured metadata and optimizing the model for deployment, we contribute a functional system capable of creative and stylistically consistent text generation.

References

Jey Han Lau, Trevor Cohn, and Timothy Baldwin. Deep-speare: A joint neural model of poetic language, meter and rhyme. Association for Computational Linguistics, 2018. URL `https://aclanthology.org/P18-1181/`.

Kai-Ling Lo, Rami Ariss, and Philipp Kurz. Gpoet-2: A gpt-2 based poem generator, 2022. URL `https://arxiv.org/abs/2205.08847`.

Zhewei Yao et al. Zeroquant: Efficient and affordable post-training quantization for large-scale transformers. *arXiv preprint*, 2022. URL `https://arxiv.org/abs/2206.01861`.