CS505 Project: Text Generation Model Comparison

Xiao Zhang
xiaozz@bu.edu

Lihao Zheng
lhzheng@bu.edu

December 22, 2023

Abstract

Our work aims to extend the application of Transformer models in text generation, building upon the foundation laid in previous homework. Specifically, our work has two parts. Firstly, we seek to identify the best model for text generation by comparing the efficacy of different Transformer models.We evaluate our results using perplexity and manual analysis of the generated sentences for assessment.

Secondly, we focus on enhancing the chosen best performing model of part 1 by experimenting with preprocessing method applied to the original dataset. The goal is to explore how preprocessing strategies can positively impact model outcomes and prove it.

Here is the github for our codes and models:GitHub link

1 Introduction

1.1 Background

Text generation is one of the most important directions of natural language processing in modern society. It can be used in many places to save people's time and help people to do creative jobs: generate massive blogs or articles, predict following words in Google Doc and "make" a strange story to give writers inspiration. After getting touched with this task from homework 5, my teammate and I had great curiosity about this work. In assignment, we had already made a preliminary work of using an LSTM model to realize

the target. However, we believe that switching to a different model (such as GPT) might yield better results. So we decided to go deeper in this topic and wanted to examine the differences in outcomes produced by different models

1.2 Methodology

Considering the diversity of directions in text generation, we made our text generation target as a fairly simple usage: Explanation. So we didn't use the data resources from nltk.corpus data package, which is composed of different literary works. The result we got after a bunch of training with different parameters on LSTM models wasn't satisfying. We turned our eyes to another famous text data resource: wikitext. The main idea of wiki is to explain a noun with more texts, so it would be the best choice if we wanted to train a model that can generate explanations. After we chose the data resource, we searched for models that can work on this text generation job. First, we studied the definition and ratings on the Internet on why they work and how to make use of them. Then we chose the correct ones for backup choices: GPT-2, T5, Big Bird, BERT and BioGPT. We tried them one by one on training the wikitext datasets. After tuning the parameters we get the final results for comparison.

1.3 Data Preprocess and Environment

First of all, we checked the origin version of wikitext. It is shown as below:
[‘’,
‘ = Valkyria Chronicles III = \n’,
‘’,
‘ Senjō no Valkyria 3 : Unrecorded Chronicles ...’]
“Valkyria Chronicles II” is the word that wiki wants to explain, using two “=” to denote it. Then one element in the list is one paragraph shown on the wiki website. If it has subsections, it will be covered by more “=” signals like: “ = = Gameplay = = \n”. So we dealt with the data by the following steps: delete the empty lines and newline; combine the explanation paragraphs to one element behind the word; Add the main word to the front of all levels of its subsections to keep content connected.

Environment of our model training: Windows 11, CPU: 12th Gen Intel(R) i7, GPU: NVIDIA RTX 3070 Ti, Memory: 16G, GPU memory: 16G

2 Result

2.1 Model Comparison

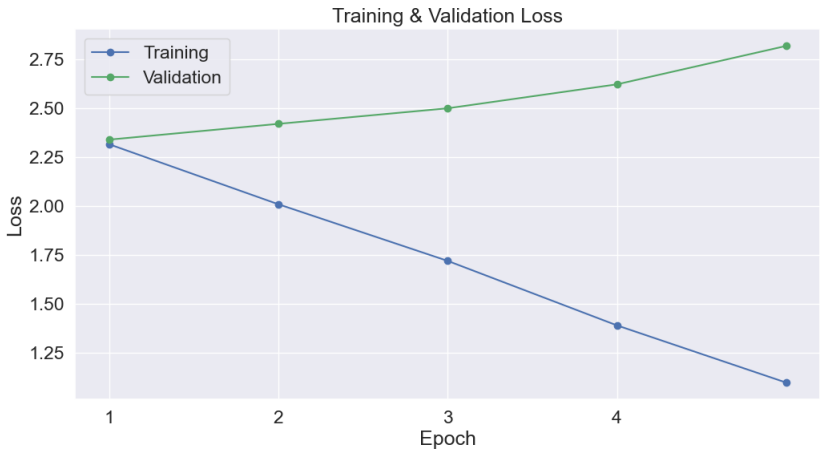

Figure 1: Loss and Validation on GPT-2 training

Perplexity: 89.5509. Text generated very well(Of course still many "AI acsent"). Here is one slice start with " = Soap = : ":
["0: = Soap = : Synthetic fragrance contains 33% sulfur, 25 % poly aromatic acid, 20 % methyl sabbat, 6 % dibenac, 20 % substituted barium, 16 %aldehyde 3-a-hydroxy-4-oxo-5-yl ketones, and 4 % astringents. In other words : 2 % will do, at most, the trick.",
"1: = Soap = : It is kept under wraps in soap formula and its contents are mixed with water. The solution, gently removed from the container, serves two purposes : directly to stimulate circulation and prevent poor results in the stomach.",
"2: = Soap = : There are various combinations of soap and soap-based products ranges from traditional mineral oils to polymers. The first is Melora and sulfide-free Combustible Essence. The brand isavailable in stores mainly in Asia and the West. Coup packages contain simple oils to mask cosmetic undertail on body and hair products."]

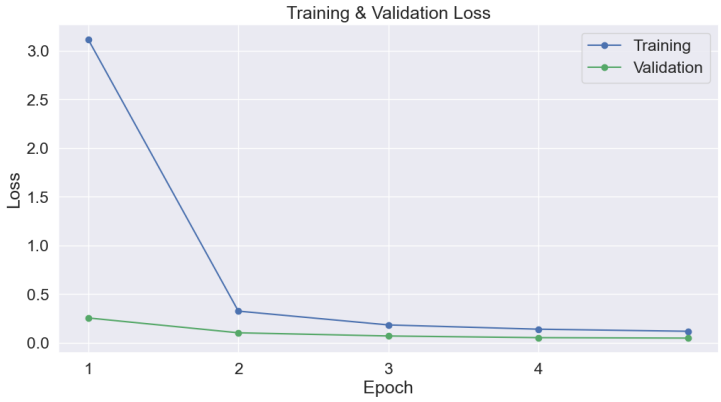

Figure 2: Loss and Validation on T5-small training

The loss value and Perplexity result(0.06) was too low and text generation was failed. This model discarded.

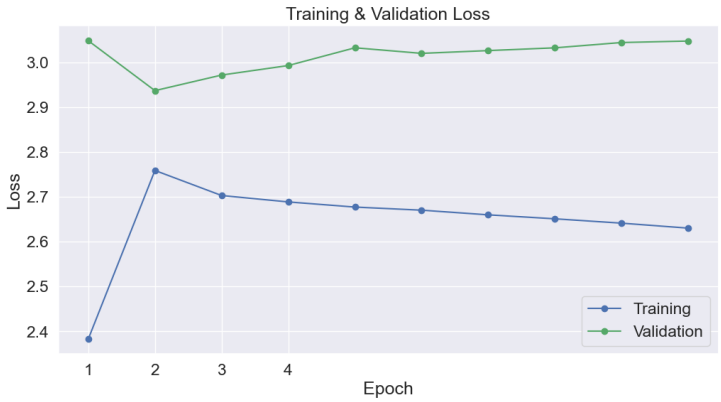

Figure 3: Loss and Validation on Big Bird training

Obviously this model failed on our data and target of text generation. Perplexity result was as high as thousands. The text generated was full of Copula and conjunctions. This model discarded

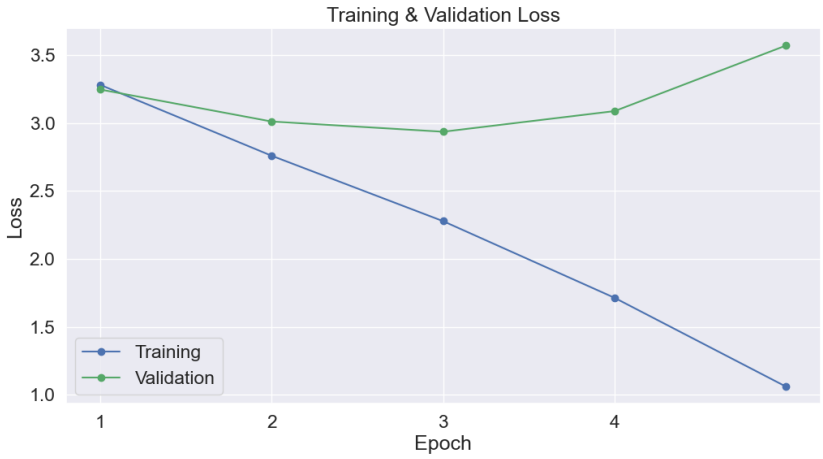

Figure 4: Loss and Validation on BioGPT training

Perplexity : 451.4723. The text generation slice does have some language logic but obviously worse than the GPT-2 model:

[“0: = Soap =: In the media, the Soap is a commercial enterprise in the United States. When viewed by figures from real-world affairs, the company was marketed as an immense commercial enterprise. The company continued in this enterprise until 2002, when it was recognized as the nation’s main executive power and commentator of the National Economy. Using sliced splinter cut, the company penetrated and joined the development of railway hotels. Soap operatives later adapted the idea of the term Spc to avoid similar problems. ”,
“1: = Soap =: Soap (SO 16) is a species of small-shaft single-shaft, common name, introduced to the British mythological Survey of India in 1874. In the Southern Way, the relative wing and tail of the adult male are identical and similar. The pelvic fins are also said to be the central Powers’ bodies. Young are to a length of 2 ft (1.8 m) and have a weight of about 30 inches (1.4 kg). ”]

2.2 Dataset Comparison

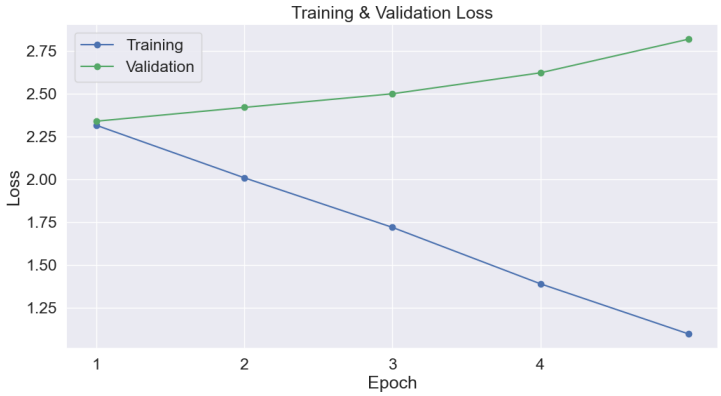

Figure 5: Loss and Validation on Preprocessed Dataset

Perplexity: 89.5509

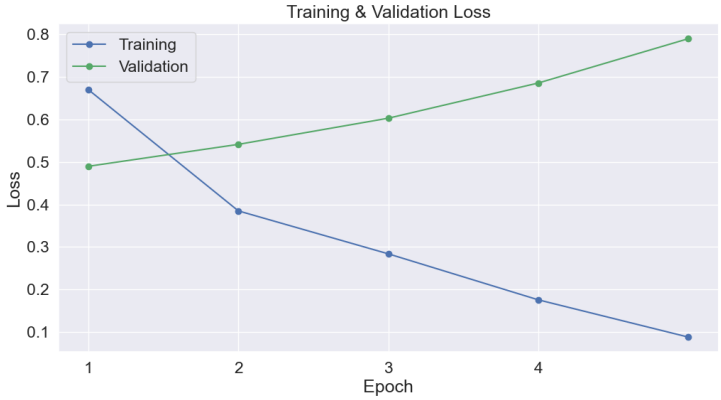

Figure 6: Loss and Validation on NOT Preprocessed Dataset

Perplexity: 1441.91. The too low loss and validation and the generated texts showed the severe overfitting problem(Copying the senetences from the origin text)
Obviously our preprocessing on data resource had a vitally positive impact on the text generation model.

3 Conclusion and discussion

3.1 Performance of different models

Based on the results of our experiments, it has been observed that among all the Transformer models we have experimented with, the GPT-2 model exhibits the best performance. Interestingly, excluding the model we show in the result section, we also attempt to use a fine-tuned BERT model for text generation. However, the results were very bad (empty result).

After checking the papers and analyzing the reasons behind models, we think that it is because the superior performance of the GPT-2 model may be attributed to its auto-regressive nature, but BERT is a model for masked language modeling and bidirectional context understanding. Therefore, Bert does not have ability to generate coherent and contextually relevant sequences of text because it can not get the causal information.

3.2 Performance of preprocessed dataset

According to our experiments, the adjusted dataset consistently outperforms the original dataset on the test set. We made modifications to the dataset, such as merging parts of its content to ensure that each sentence contains a complete causal relationship instead of the scattered nature of the original dataset. However, this has brought about certain challenges, such as causing individual sentences to become excessively long, leading to issues like prolonged processing times or insufficient RAM.

3.3 Potential limitation

Due to the extremely time-consuming and ram problem of model training, we have employed some methods to address these issues. One approach involves initially training the model on a smaller dataset to obtain a demo model. We then use the performance of this demo model to estimate its performance on the larger dataset. However, in some cases, this approach may not be effective and can even lead to adverse effects. For instance, parameters tuned honestly on the small dataset might result in severe overfitting when applied to the large dataset. Unfortunately, we have yet to find a satisfactory solution to this challenge.

4 Team Contribution

Xiao Zhang: Work on the codes of Part 2 models, contribute to the final report, and participate in team discussions on data preprocessing and other left works
Lihao Zheng: Work on the codes of fine-tuned model, contribute to the final report, and participate in team discussions on data preprocessing and other left works

5 Citation

- What is Text Generation? -HuggingFace
- GPT-2 -HuggingFace
- Text generation strategies -HuggingFace
- Datasets: Wikitext -HuggingFace
- Junyi Li, Tang, Zhao: *Pretrained Language Models for Text Generation: A Survey*

  and etc.