A Text Generation and Prediction System: Pre-training on New Corpora Using BERT and GPT-2

[1] Yuanbin Qu, [2] Peihan Liu, [1] Wei Song, [1] Lizhen Liu*, [1] Miaomiao Cheng
[1] *Information and Engineering College, Capital Normal University, Beijing 100048, P. R. China*
[2] *AIEN College, Shanghai Ocean University, Shanghai 201306, P. R. China*
{ybqu & lzliu} @cnu.edu.cn

Abstract—**Using a given starting word to make a sentence or filling in sentences is an important direction of natural language processing. From one aspect, it reflects whether the machine can have human thinking and creativity. We train the machine for specific tasks and then use it in natural language processing, which will help solve some sentence generation problems, especially for application scenarios such as summary generation, machine translation, and automatic question answering. The OpenAI GPT-2 and BERT models are currently widely used language models for text generation and prediction. There have been many experiments to verify the outstanding performance of these two models in the field of text generation. This paper will use two new corpora to train OpenAI GPT-2 model, used to generate long sentences and articles, and finally perform a comparative analysis. At the same time, we will use the BERT model to complete the task of predicting intermediate words based on the context.**

Keywords- language model; text generation; OpenAI GPT-2; BERT

I. INTRODUCTION

Now, text generation has a wide range of applications in machine translation, dialogue, abstracts, content processing, language modeling, etc. At the same time, there are many different methods of text generation, such as the Recurrent Neural Network (RNN) or Long Short-Term Memory Model (LSTM)[1], and the Encode-Decode method[2] and the sequence to sequence model[3]. The former generates a sentence forward based on a given starting word. The latter encodes the sentence as a fixed vector firstly, and then uses the vector to generate the sentence. However, with the popularity of the Transformer, language models such as OpenAI GPT-2[4] and BERT[5] also perform well in text generation tasks. We can train these models with different corpora to complete different tasks.

In this paper, we will use currently popular pre-trained models to complete some Chinese text generation tasks, including long sentence generation and masked word generation prediction task. At the same time, we provide a more intuitive website interaction page for testing, and finally a simple analysis of the generation effect.

For pre-trained models, we have selected the currently popular OpenAI Gpt-2 and BERT pre-trained models. Among them, the GPT-2 model has surpassed the previous best level in many natural language processing tasks, especially in terms of text generation. We input the beginning of the article manually, then the model automatically generates fixed-length text. Correspondingly, we will manually replace single or multiple words in the sentence with mask labels, and then use the BERT model to generate predictions for the corresponding mask label positions. For each prediction, we will provide multiple generated results.

For the corpora, we have collected a large number of Chinese corpora for training. These corpora contain a large amount of BaiduBaike information and Chinese essays, Then we will further filter the data separately to ensure the standardization of the corpora.

II. RELATED WORK

Text generation applications have evolved from early intelligent question answering and dialogue, machine translation and other systems to areas such as automatic news writing. At the same time, the corresponding technology has also made many breakthroughs from RNN, LSTM and the sequence to sequence model, all the way to the appearance of GPT-2 and BERT language models based on Transformer.

The OpenAI GPT-2 model is built using a transformer module, which is a decoder-only module. The model takes a predefined starting token as input and outputs only one token at a time. After each new token is generated, the token is added after the previously generated token sequence, and this sequence will become the new input for the next step of the model. We just need to specify the length of the build or the end flag to complete the build task. Then the model ends the generation of the text according to the predefined length or flag, and finally outputs a generation sequence.

In contrast, the BERT model is constructed by a transformer encoder module, which is a bidirectional encoder representation based on Transformer. BERT's Masked Language Model (MLM) task will predict the words covered by the mask tag

978-1-7281-6313-0/20/$31.00©2020 IEEE

according to the surrounding words. These masked words are represented by a special token $[MASK]$.

Both of these can complete certain text generation tasks, especially GPT-2, and our paper will also be based on these two models.

III. MODEL

We used OpenAI GPT-2 and BERT models to complete the generation task.

A. OpenAI GPT-2

When using this model, the main task is to generate sentences based on the starting words. The main idea of the model is to generate the next word in a loop based on an input sequence of known text until it encounters a specified length or an end identifier. Uses a single character sequence as the model input, generating a single character at a time as the output.

First, we use the BERT tokenizer to mark the sequence as a vector $\{x_t\}$, (t = 1, ..., T), where T is the length of the sequence. Then, train GPT-2 model with the vector $\{x_t\}$ as input, and record the model output sequence as $\{output_i\}$, (i = 1, 2, ...).

Then filter all tokens in $\{output_i\}$ whose probability is less than the top k tokens and whose cumulative probability is higher than the threshold, the formula is as follows:

$$logit_i = \text{Filter}(\text{output}_i, k, p) \quad (1)$$

where k is threshold of the top k tokens to be filtered, p is thresholds filtered by and cumulative probability. $\{logit_i\}$ is the probability of the next char for all char in dictionary.

Finally, we can use the multinomial method to process $\{logit_i\}$ to get the char we need.

B. BERT

When using this model, the main task is to generate predictions for the$[mask]$ label position in a given sentence. The model can make predictions based on the words around the label.

Similarly, we use a BERT tokenizer to mark the sequence as a vector $\{x_t\}$, (t = 1, ..., T), where T is the length of the sequence. Then, train the BERT model with the vector $\{x_t\}$ as input, and record the model output sequence as $\{output_i\}$, (i = 1, 2, ...).

To get the probability of the next char, we use a linear layer,

$$y_i = Softmax(\text{output}_i) \quad (2)$$

to process the $\{output_i\}$, the output of BERT. $\{y_i\}$ is the probability of the next char for all char in dictionary. We will take the top five for comparison.

IV. EXPERIMENT

A. Data

We collected a large amount of data as a model pretrained data set and stored it as two parts. The two corpora fragments are from BaiduBaike(https://baike.baidu.com/) and the LeLeKeTang website(http://www.leleketang.com/). Then, after filtering the short data that may affect the generation effect, we finally get two corpora of 5.3G and 848M in size, which will be called BaiduBaike and LLKT respectively in this paper.

TABLE I. EXAMPLE OF CORPORA

	BaiduBaike	**LLKT**
Title	对联 Couplets	开心的一天 Happy day
Content	对联，又称楹联或对子，是写在纸、布上或刻在竹子、木头、柱子上的对偶语句言简意深，对仗工整，平仄协调，是一字一音的中文语言独特的艺术形式 ... (Couplets, also known as couplets or couplets, are antithetical sentences written on paper, cloth or carved on bamboo, wood or pillars, which are simple and profound, neat and harmonious,and are unique artistic forms of Chinese language with one word and one tone ...)	星期五的下午，我早早的回到家把作业做完。因为明天爸爸带我去弟弟家，我兴奋的不得了。所以我就早早把作业都写完了，等待明天的到来 ... (On Friday afternoon, I came home early to finish the homework. I will be very excited because my dad will take me to my brother's house tomorrow. So I finished my homework early and waited for tomorrow's arrival ...)

Although the length of the filtered articles still varies widely, the consistency of each article is basically guaranteed. Table I shows sample data examples in the data set (because of the length, we only intercept some of the articles for display, and for the convenience of reading, we also provide English translations of the examples). Each instance contains a title and content.

B. Training

There are two models, GPT-2 Model and BERT Model, we introduced in the section III.

Firstly, to complete long sentence generation task, we will train the GPT-2 model with the BaiduBaike and LLKT corpora.

For better training of Chinese, we choose char-level to split sentence. Every Chinese character has rich meanings, and the number of commonly used Chinese characters is about 5,000. But the number of commonly used Chinese is too much. Then we divide the corpora into 100 parts.

For model parameter settings, we set the learning rate to 1.5e-4, bacth_size to 8, and default the other parameters. We will train the model for 5 epochs. When training is complete, we save the final model to complete the sentence generation task.

For masked word generation prediction task, we will use the bert-base-chinese model.

C. Generating

For long sentence generation tasks, we choose keywords that often appear in the training corpora to generate sentences.

The generation algorithm as fellow:

Input: keyword x, model G, length

Output: sentence s

s ← x

for i = len(x) to length do

G.init()

next char ← G(s)

s ← s + next char

Endfor

return s

The keyword x is a known starting word, the model G is GPT-2 model, the *length* is the set length of the result sentence. The sentence s is the sentence fed to the model and at the final it is the result sentence. So we define the input as the form $\{x, G, length\}$.

At each time step, the model G is initialized and predicts the most likely token as next token through the input sentences. Then the input sentence s is concatenated with the next token, and will be fed to the model G next time step. Until the generation length of the model G reaches the set length, the generation process will stop, and the final output sentence is the result.

For the mask generation task, we choose some shorter sentences to generate the prediction. The generation algorithm as fellow:

Input: sentence s, model B

Output: predicted word $\{w_t\}$

$\{w_t\}$ ← B(s)

return $\{w_t\}$

the sentence s is a known sentence with one or more mask labels, model B is the BERT model, and $\{w_t\}$, (t = 1, 2, ..., T) is the final predicted word, where T is the number of mask label in the sentence. So we define the input as the form $\{s, B\}$.

We input the sentence s with one or more mask labels into model B, and the model generates the predicted words corresponding to the mask position. We take the top five tokens of the prediction score as the final generated reference.

D. Website

At the same time, we also provide a website to make us more intuitive and convenient for testing and comparison. We finish the website with Django, which is an excellent web framework for the python language to complete the writing of the page.

V. ANALYSIS

Given a starting word or part of a sentence, everyone can write a completely different sentence. The diversity of sentence makes automatic evaluation metrics impossible. Therefore we only perform manual evaluation in this study.

TABLE II. EXAMPLES GENERATED BY GPT-2 MODEL

	Keyword	Sentence
BaiduBaike	中国 (China)	**example1** 中国人民解放军的第 22 军，第 134 师（司令员兼第 60 军军长）。1946 年 10 月 10 日，第 71 军（司令员兼政治委员）在江南召开军事训练会议，决定成立江南抗日民主政府， **example2** 中国科学院生物技术研究所研究员、中国科技大学博士生导师。主要学术成果：“中国生物信息处理及其应用技术的研究”获国家科技进步三等奖。1965 年以来，先后主持过国 **example3** 中国科学院研究生院博士研究生部。主要研究方向为中国地质学，地质科学，地球科学，地理信息系统。1957 年毕业于北京地质学院地球物理系。历任北京地质学院、中国岩石研究
LLKT	今天 (Today)	**example1** 今天，我们就在这里举行一场别开生面的“开学第一课”。这次活动是由一年级的小记者和二班的小记者组成的。我们在这次活动中，我们也收获了无数的知识，也收获了无数的快乐。 **example2** 今天的生活很精彩。今天，我在上课的时候，发现了一只小狗和一对母子，母子俩的家在一起，我非常好奇，便走过去问它：“小狗，你在干什么？” **example3** 今天，老师把我们分成四个人，每个人都拿出一张纸，每个人都拿出一张纸，我也拿来一张，就是用来写一些自己的日志。

TABLE III. EXAMPLES GENERATED BY BERT MODEL

Sentence	Predicted word(Probability)
海内存知己， [MASK]涯若比邻。	天(0.254) 空(0.141) 蓝(0.124) 国(0.601) 入(0.212)
你[MASK]， 我[MASK]小[MASK]。	好(0.849) 是(0.372) 孩(0.156) 说(0.305) 叫(0.337) 明(0.135) 看(0.255) 的(0.398) 宝(0.135) 是(0.074) 爱(0.378) 强(0.125) 们(0.063) 和(0.153) 白(0.120)

In the paper, we use manual ratings on two important metrics in generation system.

- Topic relevance: on how much is the output sentence consistent with the given starting word, and is the meaning of the output sentence appropriate.
- Readability: on how much is the output sentence readable, and grammatical.

In Table II, there are some results generated from starting word by GPT-2 model. There are different generation examples of different corpora.

Generally, the meaning and readability of most of the generated sentences are appropriate, but when we continue to generate on the basis of the first generation, there will be duplicates. At the same time, compared to a LLKT, the result of BaiduBaike is worse.

Table III shows the results generated using the BERT model. Whether for single or multiple mask labels, the predictive words generated by the model show good results in readability and coherence.

From the experimental results, we have the following observations.

- In terms of long sentence generation, the GPT-2 model has a better performance, but the effect is slightly worse when it continues to generate downwards based on the generated long sentences. This may be due to the inconsistent sentence lengths in the training corpora, which needs further modify.
- In terms of masked word predictions, the BERT model generated more results than expected. However, when the predicted words are some common collocations, there is no significant difference in prediction scores.

There is still a probability that we can not get the appropriate sentence. One reason may be that our corpora is not standardized, so the model cannot learn all the language rules. Another reason is that models are more sensitive to unspecific corpora. Our corpora contains a large amount of BaiduBaike data and essays, but the length and style of these sentences are still very different. Mostly, there will be duplicates when generating longer text.

For future research, we will try to use the model to do more tasks. We will further process the corpora. According to the existing model, design a better effect generation system.

VI. Conclusion

We have pre-trained the model by using different corpora, and used the trained model to complete the long sentence generation and masked word generation prediction task. The former mainly generates sentences by looping down from the start word, and the latter is based on the surroundings word to generate intermediate words. Through the experimental results, we can know that the GPT-2 and BERT models perform very well in text generation tasks. However, there are still some shortcomings, such as readability, corpora data, and training methods, which may cause generated sentences to be repeated, etc. In the future, we will try to find some ways to solve this defect.

Acknowledgment

We would like to thank anonymous reviewers for their helpful suggestions and particularly the annotators for their contributions on the dataset. The research work is funded by the National Natural Science Foundation of China (No.61876113), the Beijing Natural Science Foundation (No.4192017) High-level Teachers in Beijing Municipal Universities in the Period of 13th Five-year Plan (CIT&TCD20170322).

References

[1] Nallapati, R., Zhou, B., Gulcehre, C., & Xiang, B. (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:1602.06023.

[2] Guthaus, M. R., Ringenberg, J. S., Ernst, D., Austin, T. M., Mudge, T., & Brown, R. B. (2001, December). MiBench: A free, commercially representative embedded benchmark suite. In Workload Characterization, 2001. WWC-4. 2001 IEEE International Workshop on (pp. 3-14). IEEE.

[3] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104-3112).

[4] Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D. & Sutskever, I.(2018), Language Models are Unsupervised Multitask Learners.

[5] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[6] Poerner, N., Waltinger, U., & Schütze, H. (2019). BERT is Not a Knowledge Base (Yet): Factual Knowledge vs. Name-Based Reasoning in Unsupervised QA. arXiv preprint arXiv:1911.03681.

[7] Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y. (2019). Mass: Masked sequence to sequence pre-training for language generation. arXiv preprint arXiv:1905.02450.

[8] Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., & Socher, R.(2019). Ctrl: A conditional transformer language model for controllable generation. arXiv preprint arXiv:1909.05858.

[9] Uchimoto, K., Isahara, H., & Sekine, S. (2002, August). Text generation from keywords. In Proceedings of the 19th international conference on Computational linguistics-Volume 1 (pp. 1-7). Association for Computational Linguistics.

[10] Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013, February). On the importance of initialization and momentum in deep learning. In International conference on machine learning (pp. 1139-1147).

[11] Guo, J., Lu, S., Cai, H., Zhang, W., Yu, Y., & Wang, J. (2018, April). Long text generation via adversarial training with leaked information. In Thirty-Second AAAI Conference on Artificial Intelligence.

[12] Alfaro, F., Ruiz Costa-Jussà, M., & Rodr´ ıguez Fonollosa, J. A. (2019). BERT masked language modeling for co-reference resolution. In Proceedings of the First Workshop on Gender Bias in Natural Language Processing (pp. 76-81).

[13] Jiang, Z., Xu, F. F., Araki, J., & Neubig, G. (2019). How Can We Know What Language Models Know?. arXiv preprint arXiv:1911.12543

[14] Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., & Bengio, Y.(2016, March). End-to-end attention-based large vocabulary speech recognition. In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on (pp. 4945-4949). IEEE