Dual Neural Network Classification Based on BERT Feature Extraction for Authorship Verification

Notebook for PAN at CLEF 2021

Xiaogang Miao, Haoliang Qi*, Zhijie Zhang, Guiyuan Cao, Ruilan Lin, Wenbin Lin

Foshan University, Foshan, China

Abstract

Authorship verification is the task of deciding whether two texts have been written by the same author. We regard authorship verification as a classification task. A dual neural network is proposed to classify the features of text extraction. Especially, BERT is exploited as the encoder to extract the text features. After training and forecasting on the given pan20-authorship-verification-training-small data set, the weighted average of the specified evaluation indexes (including: AUC, F1, c@1, f0.5u, Brier) can reach about 0.85.

Keywords

Dual Neural Network, BERT, Long Text Classification

1. Introduction

Authorship verification is an active research field in computational linguistics. By comparing the writing style of text, the author determines whether the same author has written two texts [1,2]. It has been widely used in the academic field. It can be used as a detection direction of academic paper fraud and plagiarism, not only detecting word repetition rate. In 2021, the training data are the same as last year, open-set Authorship Verification. It is difficult to see if there are new authors and themes, so the task difficulty has been increased [3]. Essentially, given two paragraphs, it is a text pair classification problem to determine whether the same author writes it and whether the label is "true" or "false". This paper uses a double-input neural network model, which can extract and learn each text information step by step and then train and predict the results in the upper neural network.

2. Datasets

The dataset used by the Authorship verification is given by the evaluation web pan@clef. The train (calibration) and test datasets consists of pairs of (snippets from) two different fanfics, that were obtained drawn from fanfiction.net. Each pair was assigned a unique identifier and we distinguish between same-author pairs and different-authors pairs [3].

There are two datasets verified by the author. The difference lies in the number of text pairs included. For larger datasets and smaller datasets [4], we use smaller datasets, which contain 52,601 pairs of text. They are all written in a JSON file, including 'id', 'fandoms', 'pair' in pair.json file, then 'id', 'same' and 'authors' in truth.json file. In fact, we mainly use the text pairs in the 'pair' item to extract the required information [3].

In the data set, most of the text characters are between 30,000 and 20,670, a few are between 30,000 and 300,000, and most of the words are between 55,433 and 42,372. There are only NYAN, HUAE and

[1] CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania

EMAIL: xiaogangmiao163@gmail.com (A. 1); qihaoliang@fosu.edu.cn (A. 2) (*corresponding author); zhangzhijie5454@gmail.com (A. 3); caoguiyuan2020@163.com (A. 3); LRLlinruilan@163.com (A. 3); LWBlinwenbin@163.com (A. 3)
ORCID: 0000-0001-6816-0922 (A. 1); 0000-0003-1321-5820 (A. 2)

other words in the text pairs of some of the data, because the data is from fanfection.net, we can't determine whether it is some wrong data.

3. Method

We adopt a dual-input neural network model to learn text information and a feature extraction processing text based on the BERT model [5]. The specific implementation method is as follows.

3.1. Dual Neural Network Classifier

We used a dual-input model for the authorship verification task for classification training. The specific model is shown below. Figure 1-(a) is the preprocessing process of the original text, and the long text is aligned with the clause. Figure 1-(b) is to put the well-divided sentence set into the BERT model to extract the corresponding feature vector of the text. Figure 1-(c) indicates that the extracted feature vectors are sent to the dual-input neural network for final prediction classification.

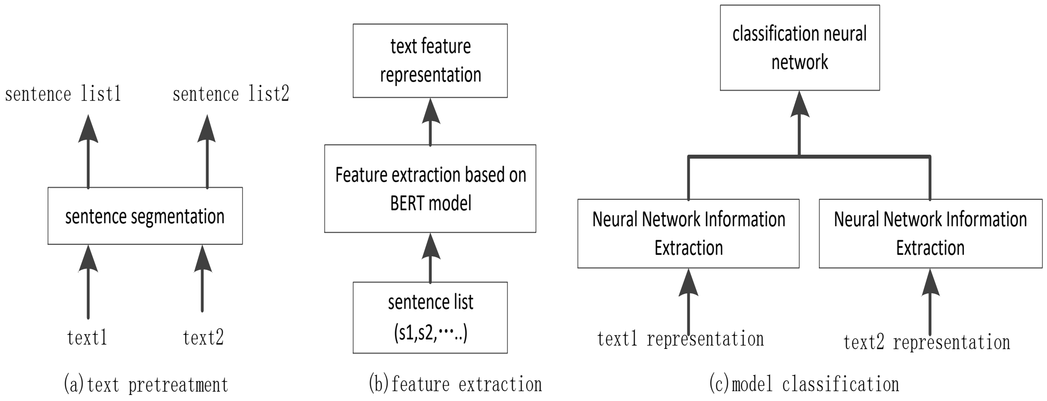
sentence list1
sentence list2
sentence segmentation
text1
text2
(a)text pretreatment
text feature representation
Feature extraction based on BERT model
sentence list (s1,s2,…..)
(b)feature extraction
classification neural network
Neural Network Information Extraction
Neural Network Information Extraction
text1 representation
text2 representation
(c)model classification

Figure 1: model summary

3.2. Text Processing

First of all, according to statistics, each text segment is based on multiple sentences, so we use punctuation marks as sentence separators to divide, mainly full stop, exclamation mark, and question mark. In this way, the long text segment can be divided into several small sentences, which can be processed by BERT. However, there are also some problems. For example, the sentences which are composed of multiple repeated words such as 'NYAN' mentioned in the data set do not have the above sentence segmentation conditions.

Some text segments are more than 20,000 characters in length, but sentence separators are not used in the whole paragraph, which may be related to the author's writing habits. Therefore, if there is no sentence after the first sentence segmentation, we need to segment the sentence directly according to the length of the character. In theory, some information will be lost. Fortunately, the sentences that need to be segmented occupy a small part of the whole dataset, which is less than 0.1%, so it has little impact on the final result.

Through the above text segmentation operation, we will divide the long text into several short sentences. Here we use BERT as a feature extractor. As a pre-training model, BERT has multiple versions. We use BERT-Large, Cased (Whole Word Masking): 24-layer, 1,024-hidden, 16-heads, 340M parameters [5]. Because the parameters have been trained in advance, in the normal text classification task, we use BERT as a feature extractor and then a downstream task (usually neural network) as a model and then fine-tune the overall performance. But in this task, we can't input all the text at one time, so we only consider using BERT as a feature extractor. The specific method is to extract the CLS vector of each short sentence as the feature of the short sentence through 12 level

Transformer. CLS obtains the sentence-level information representation through the self-attention mechanism, which can capture the context information representation in the current environment. Then the CLS vectors of several short sentences are superimposed to represent the sentence feature vectors of long text.

4. References

Based on the above general introduction, we will introduce the working model and results in detail below.

4.1. Second level heading

Because of the length of the text in the task data set and the information contained in the text is very scattered, it is difficult to use sliding window truncation. However, since the computation resources and time consumed by BERT increase with the square level of token length, it can't handle too long tokens. At present, it only supports 512 tokens at most. If the token is too long, it is easy to overflow memory. Therefore, we need to design a clever method to solve this problem when using BERT to process long text. Therefore, we need to first make some clauses in the text. First of all, we divide the data set into two parts: the positive and negative samples of the data set are 50%, the first half is true, and the later version is false. Then we take 52,000 as the training set and the rest as the verification. The first method is to splice the feature vectors of two texts because each feature vector is 768 dimensions, and the spliced vector is a feature representation of 1,536 dimensions. Then we send this feature into a simple neural network for binary classification, but after training, we find that its accuracy in the training set is not very high. The overall verification set is only 0.82. Then the second method is that we build a double input neural network, which is the fusion of the information from the training of the two text features in their respective neural networks [6]. After a two-classification prediction, the final overall on the test set can reach 0.87. We speculate that the reason for this may be that the neural network can not effectively recognize the whole order of sentences in the first step after using splicing, or we have not trained it sufficiently. In fact, the number of neurons and activation function, and other parameters are selected and experimented with according to our experience. The implementation method is not necessarily the optimal result under this idea but only a reference scheme.

4.2. Second level heading

In addition to the above data allocation, we also allocate 80% of the training data and 20% of the validation data to compare the training before starting the formal training to find the appropriate number of training epochs. Table 1 shows the experimental results.

Table 1

The score of training data and validation data after processing

data set	AUC	C@1	F0.5U	F1-SCORE	OVERALL
training	0.866	0.938	0.858	0.865	0.881
verification	0.855	0.914	0.839	0.857	0.867

Because the final score on the test set was not given at the time of final submission, we can only show the score on the training set now, which may not be very accurate.

5. References

In the authorship verification task, we use the dual-input neural network model to accept the text features extracted from the BERT model for classification learning. In the model, the feature information of the two texts can be learned respectively and then sent to the upper neural network for classification. To some extent, it avoids the decrease of learning efficiency caused by the fact that the splicing cannot learn the two-sentence sequences. Good results are achieved on the training dataset given by the evaluation task. Note that we did not fine-tune the native model when we used BERT as a feature extractor. Because we first segment the text and then extract features, if you want to fine-tune should be considered into the segmented sentence for operation. However, the information obtained by this model may be partial, incomplete information based on the whole text, and we are not sure whether this will help on open sets. It is hoped that someone can solve the problem of feature extraction accuracy in the future.

6. References

[1] Kestemont M, Manjavacas E, Markov I, et al. Overview of PAN 2021: Authorship Verification, Profiling Hate Speech Spreaders on Twitter, and Style Change Detection[J].

[2] Bevendorff J , Chulvi B , GLDLP Sarracén, et al. Overview of PAN 2021: Authorship Verification, Profiling Hate Speech Spreaders on Twitter, and Style Change Detection: Extended Abstract[M]. 2021.

[3] Organization(2021),https:/pan.webis.de/

[4] Rong X , Wang X , Yan L , et al. Research and application on improved BP neural network algorithm[C]// Industrial Electronics & Applications. IEEE, 2010.

[5] Devlin J , Chang M W , Lee K , et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. 2018.

[6] Boenninghoff B , Rupp J , Nickel R M , et al. Deep Bayes Factor Scoring for Authorship Verification[C]// Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. 2020.