Analysis of Irony and Stereotype Spreaders Based On Convolutional Neural Networks

Notebook for PAN at CLEF 2022

Dong Yuan, Wenyin Yang*, Li Ma, Zhenlin Tan, Qidi Lao

Foshan University, Foshan, China

Abstract

In order to solve the task of Profiling Irony and Stereotype Spreaders on Twitter released by PAN 2022 organizers, this paper proposes a two-layer convolutional neural network based on deep learning. This task is essentially a binary classification task.Due to its strong feature extraction ability, the convolutional neural network has been very active in the classification tasks of professional competitions[1]. The proposed model achieves an accuracy of 0.8 and a loss of 0.3 on the training set. The running time of the model is only 2 minutes.

Keywords

Irony and Stereotype Spreaders, Convolutional Neural Networks, Binary Classification

1. Introduction

With the rapid development of the Internet, more and more people are used to expressing their opinions on the Internet. Now the Internet is full of all kinds of speech, there are positive speech, negative speech, hate speech and so on. It is very meaningful to determine what category the speech on the Internet belongs to. Therefore, the Profiling Irony and Stereotype Spreaders on Twitter Task held by PAN 2022, the main purpose of the competition is to judge whether the Twitter user is Irony and Stereotype Spreaders. This paper proposes a model based on convolutional neural network to achieve this purpose[1]. First, the model preprocesses the original data, including splitting at spaces, removing punctuation, and segmenting words. Data preprocessing maps each word to a positive integer. Next, the model maps the positive integers into a multidimensional vector, which is used as the input to the convolutional neural network. The first layer of the model's convolutional neural network performs preliminary feature extraction on the input tensor, and then inputs the result to the pooling layer for further feature extraction. In order to extract more important semantic features, the model stacks a convolutional neural network and a pooling layer. Finally, classification is performed through a fully connected layer.

2. Dataset

The datasets provided by the Profiling Irony and Stereotype Spreaders on Twitter task are mainly derived from the speeches made by Twitter users. The training set contains 420 xml files named after the user ID—each XML is related to a single author—containing 200 comments made by the user on Twitter. In addition, it contains a truth.txt file, which also contains 200 pieces of information, and each piece of information is marked with an ID corresponding to the label "I" or "NI". The test set contains

180 Twitter users, and each user has a different number of comments Table 1 shows the analysis of the training set and test set.

Table 1
The detail of Profiling Irony and Stereotype Spreaders on Twitter Task

Dataset	Number of users	Number of tweets	Number of labels I	Number of labels NI
Training set	420	84000	42000	42000
Test set	180	-	-	-

3. Model Architecture

The model framework proposed in this paper is shown in Figure 1. The model is divided into two modules: text preprocessing and feature extraction. The role of the text preprocessing module is to perform data format conversion, standardization, word segmentation, and vectorization of the original data. The output of this module is a bunch of multidimensional tensors for easy processing by the feature extraction module. The function of the feature extraction module is to extract the semantic features in the text sequence and output the predicted result. Semantic feature extraction is divided into three stages. The first stage is preliminary extraction, which is done by Convolutional Neural Network Layer and Max Pooling Layer. The second stage is depth extraction, which is done by Convolutional Neural Network Layer and GlobalMax Pooling Layer. The third stage is the prediction result, which is done by Fully Connected Layer.
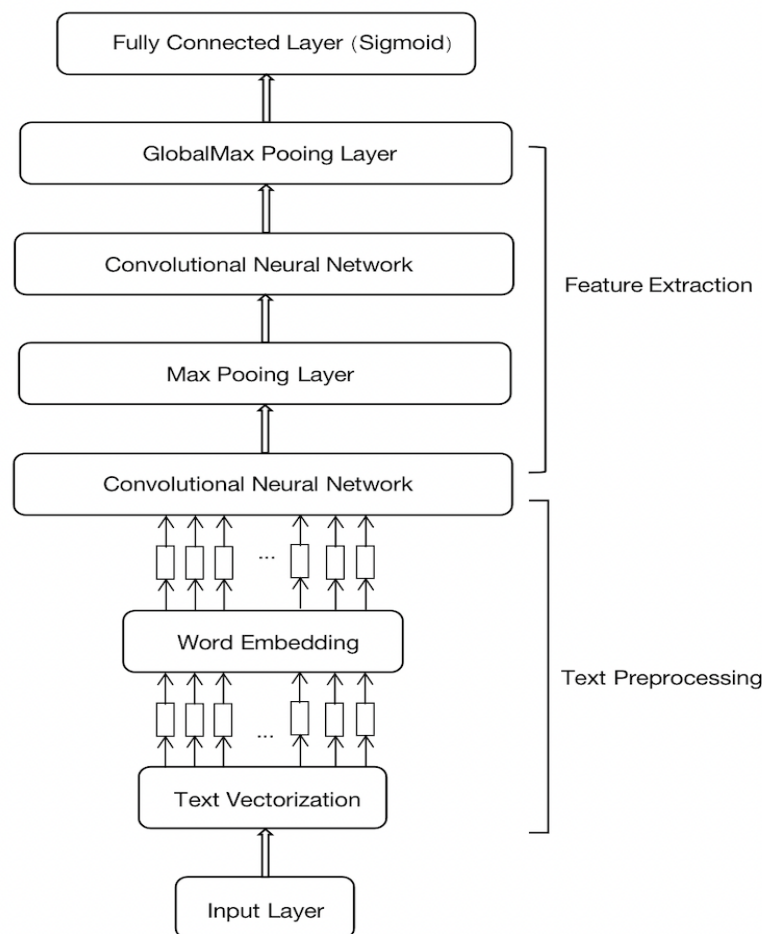
Figure 1: Model architecture

For binary classification tasks such as Profiling Irony and Stereotype Spreaders on Twitter, using a combination of 1D convolutional neural networks and pooling layers can get a fairly good accuracy, and the model is less computationally expensive[4]. Next, the paper will introduce the role of each layer of the model in detail.

3.1.　　Text preprocessing

The first step of text preprocessing is to convert the XML format dataset into pure text. According to the label "I" or "NI" corresponding to each author id in the truth.txt file, the model stores the 420 xml files in the newly created folder "I" or "NI" in txt format, respectively[5]. At the same time, the model will read the information in each file. The above work is done by Input Layer. This is convenient for subsequent word segmentation and mapping of the text into integers.

The model uses the Text Vectorization function in the Tensorflow framework to process text. It lowercases the text, splits it on spaces, strips punctuation, and outputs an integer vocabulary index[6]. The output result of this layer is a sequence of 9406 positive integers. Each number corresponds to a word, and finally a dictionary can be obtained, where the key is the word and the value is the positive integer corresponding to the word. At the same time, the maximum length of the text processed by the model through the Text Vectorization Layer is 9406. The text larger than this length will be truncated, and the text smaller than this length will be filled[7].

Because the neural network cannot handle integer sequences directly, the model inputs the result of Text Vectorization into the Word Embedding Layer and maps it into a dense vector. The Word Embedding Layer is equivalent to a dictionary that takes integers as input, looks up those integers in the internal dictionary, and returns the associated vector. After training and testing multiple models, we determined that the model is better when the parameter embedding-dimensionality of the Word Embedding Layer is 130.

Above, we have completed the preprocessing of the text. Next, feature extraction will be performed on the preprocessed text.

3.2.　　Feature extraction

In fact, in order to determine the most suitable hyperparameters (batch size, optimizer, filter size, etc.) and model structure, we conducted multiple tests and repeatedly compared the loss value, accuracy and model calculation cost of each test result. The resulting hyperparameters are detailed in the introduction to each layer of the model.

For lightweight text sequence processing, this paper selects a one-dimensional convolutional neural network and pooling layer for feature extraction 错误!未找到引用源。. The output of the Word Embedding Layer is (1,9406,130), which is a three-dimensional tensor. The first dimension represents the number of data files processed each time, the second dimension represents the maximum length of the text sequence, and the third dimension represents the Word Embedding Layer output dimension. In order to enhance the ability of model feature extraction, we adopt a two-layer one-dimensional convolutional neural network and a two-layer pooling network.

The output dimension of the first Convolutional Neural Network Layer is 128, and the length of the convolution window is 72, that is, the convolution window scans 72 words each time.The result output by the Word Embedding Layer will get a matrix of size 9335*128 through the first first Convolutional Neural Network Layer. In order to reduce the size of the output data, the output three-dimensional tensor (1,9335,128) is input to Max Pooling Layer. The role of the Max Pooling Layer is to downsample the feature text sequence and output the maximum value of each channel. The output matrix size of this layer is 583*128.

So as to further extract the main features in the text sequence, we added a second Convolutional Neural Network Layer (the dimension of the output space is 64, the length of the convolution window is 36), and the output matrix size of this layer is 548*64. This matrix is then passed through the GlobalMaxpooling Layer to extract important semantic information in the text sequence, and the output result is a matrix of size 1*64 错误!未找到引用源。.

At the end of the model, the model stacks the Fully Connected Layer, and the activation function adopts the Sigmoid Function to determine whether the speech information in the input file is I or NI.

4. Results

To evaluate the model proposed in this paper, we first learn from 84,000 tweets given by PAN. Among them, the epoch is set to 7, and the value of each epoch is shown in Table 2. We tried to set the epoch to be larger, but when the epoch was greater than 7, the model performance did not change much.

Table 2
Training set running results

Metrics	Epoch=1	Epoch=2	Epoch=3	Epoch=4	Epoch=5	Epoch=6	Epoch=7
Accuracy	0.5594	0.7021	0.7606	0.7783	0.7836	0.7949	0.8143
Loss	0.6629	0.5749	0.5100	0.4800	0.4471	0.4182	0.3918

To further evaluate the performance of the model, we randomly split the 420 xml files given by PAN as a training set into two parts, of which 320 xml files are used as a new training set and the remaining 100 xml files are used as a test set. According to the result labels output by the comparison model and the labels of these 100 author ids given by PAN, we found that the accuracy of the test set reached 0.85.

Finally, the running accuracy of the proposed model on the test set given by PAN2022 reaches 0.75.

5. Conclusion

This paper proposes a model that utilizes a two-layer convolutional neural network and a two-layer pooling network to solve the Profiling Irony and Stereotype Spreaders on Twitter Task in PAN@CLEF 2022. Semantic features can be extracted from text sequences using convolutional neural networks. The two-layer convolutional neural network can further enhance the effect of feature extraction. Applying the model to the PAN 2022 Profiling Irony and Stereotype Spreaders on Twitter Task for binary classification can not only obtain high accuracy, but also has a low computational cost, fast running speed, and strong practical value.

Experiments show that for lightweight and simple tasks such as text classification and temporal prediction, small one-dimensional convolutional neural networks can replace other complex networks such as recurrent neural networks.

6. Acknowledgment

7. References

[1] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pezik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle. Overview of PAN 2022: Authorship Verification, Profiling Irony and Stereotype Spreaders, and Style Change Detection, in: A. B. Cedeno, G. D. S. Martino, M. D. Esposti, F. Sebastiani, C. Macdonald, G. Pasi, A. Hanbury, M. Potthast, G. Faggioli, N. Ferro (Eds.), Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), Springer, 2022.
[2] Ortega-Bueno R., Chulvi B., Rangel F., Rosso P. and Fersini E.Profiling Irony and Stereotype Spreaders on Twitter (IROSTEREO) at PAN 2022.In: CLEF 2022 Labs and Workshops, Notebook Papers, CEUR-WS.org.

[3] M Siino, E Di Nuovo, I Tinnirello, M La Cascia:Detection of hate speech spreaders using convolutional neural networks,CLEF, 2021 - ceur-ws.org.

[4] Y. Kim, Convolutional neural networks for sentence classification, arXiv preprint arXiv:1408.5882 (2014).

[5] M. Potthast, T. Gollub, M, Wiegmann, B. Stein: TIRA Integrated Research Architecture, in: Information Retrieval Evaluation in a Changing World, ser. The Information Retrieval Series, N. Ferro, C. Peters, Berlin Heidelberg New York: Springer, Sep. 2019.

[6] Chollet, Francois. Deep learning with Python. Simon and Schuster, 2021.

[7] Reyes A., Rosso P. On the Difficulty of Automatically Detecting Irony: Beyond a Simple Case of Negation. In: Knowledge and Information Systems, 40(3):595-614, 2014.

[8] F. Rangel, A. Giachanou, B. Ghanem, P. Rosso, Overview of the 8th author profiling task at pan 2020: Profiling fake news spreaders on twitter, in: CLEF, 2020, p. 1.