Authorship Verification Using Convolutional Neural Network

Notebook for PAN at CLEF 2022

Yihui Ye, Han Y*, Zeyang Peng, Mingjie Huang, Leilei Kong, Zhongyuan Han

[1] *Foshan University, Foshan, China*

Abstract

The PAN@CLEF 2022 Authorship Verification[8] is the task of determining if two texts are written by the same author based on comparing the texts' writing styles. In this work, we propose a model that blends Bert and TextCNN[6] features and raises a data reorganization method to increase model training data to improve performance.

Keywords

Pre-trained model, TextCNN, Data reorganization, Authorship verification

1. Introduction

Authorship verification can be applied to plagiarism detection, paper duplication checking, etc. Authorship verification is a category of text classification tasks, which can be regarded as a fundamental problem of NLP, that is, to determine whether the same author writes two texts. The work presented in this paper was developed as a solution for the Authorship verification task of the competition PAN@CLEF 2022. Unlike the previous year, this year's task dataset contains four types of text: essay, text_message, email, and memo.

Different from the previous year, the overall distribution of this year's texts tends to be short. We try to blend the Pre-trained model Bert, which has performed well in natural language processing, with TextCNN, which is suitable for dealing with a short text. Our work proposes an authorship verification model based on Pre-training and TextCNN. Firstly, we used a pre-trained model to initially extract the author's writing style features of the text. Secondly, we used the TextCNN to extract the texts' style features and complete the authorship verification task. Meanwhile, we propose a data reorganization method to obtain a large amount of training data. The model is trained through a more enormous amount of training data to improve the judgment ability of the model further.

2. Datasets

The PAN@CLEF 2022 Authorship verification task dataset contains 12,264 text pairs and labels. The meaning of the label is whether the two texts are written by the same author. If it is, it will be 1. If not, it will be 0. This year's texts consist of four types, essay, text_message, email, and memo. There are six combinations of text pairs, and their distribution types and quantities are shown in Table 1.

[1]CLEF 2022 – Conference and Labs of the Evaluation Forum, September 5-8, 2022, Bologna, Italy

EMAIL:oldsport996@gmail.com(A. 1); hanyong2005@fosu.edu.cn(A. 2)(*corresponding author); pengzeyang008@163.com(A. 3); mingjiehuang007@163.com(A. 4); kongleilei@fosu.edu.cn(A. 5); hanzhongyuan@fosu.edu.cn(A. 6);
ORCID: 0000-0002-7369-7537 (A. 1); 0000-0002-9416-2398 (A. 2); 0000-0002-8605-4426 (A. 3); 0000-0002-0889-5027 (A. 4); 0000-0002-4636-3507(A. 5); 0000-0001-8960-9872(A. 6)

Table 1
The detail of the PAN@CLEF 2022 Authorship verification task dataset

Text1 type	Text2 type	Quantity
essay	text_message	1,182
essay	email	1,618
essay	memo	186
email	text_message	7,484
memo	text_message	780
memo	email	1,014

This year, the authorship verification task focused on the scene of the open domain set and applied cross-text type recognition.

3. Method

3.1.Dataset Preprocessing

All the texts in the training data contain a total of 56 authors. Each author has written hundreds of texts, by collecting all the texts written by each author, recombining these texts and assigning labels to generate new training samples.

The texts with the same author label are placed in the same list, and a total of 56 lists are established. The texts in each list are connected in pairs to obtain training samples with a label of 1. The texts in different lists are combined in pairs to get training samples with a label of 0.

Suppose list1=['en_xx', $text_{11}$, $text_{12}$, . . . , $text_{1n}$], where $text_{1n}$ is the final text of the first author we have recollected. We can get new training samples like [$text_{11}$, $text_{12}$, '1'], [$text_{12}$, $text_{13}$, '1'], et al.

Suppose list2=['en_yy', $text_{21}$, $text_{22}$, . . . , $text_{2n}$]. We can get new training samples like [$text_{11}$, $text_{21}$, '0'].

The texts of different authors are much more than the text pairs of the same author. When expanding the training data, we adopt different text pairing strategies. Firstly, we pair all the texts under each author to obtain samples of the same author. The first text of the same author is paired with the second text, and the second text is paired with the third text until the penultimate text is paired with the last text. Secondly, we pair the first text of the first author with the first text of the second author, the first text of the first author with the first text of the third author, until the first author and the first text is matched with the last The first text of one author is paired, then the first text of the second author is paired.The first text of the third author and so on, until the first text is matched in a loop. In the same circle, the second text of the first author is matched with the second text of the third author. For the text pairs generated by different author combinations, we apply the first 17 texts of each author to combine. For the specific combination strategy, please refer to the figure below. The figure below illustrates the process of matching text pairs with six authors.

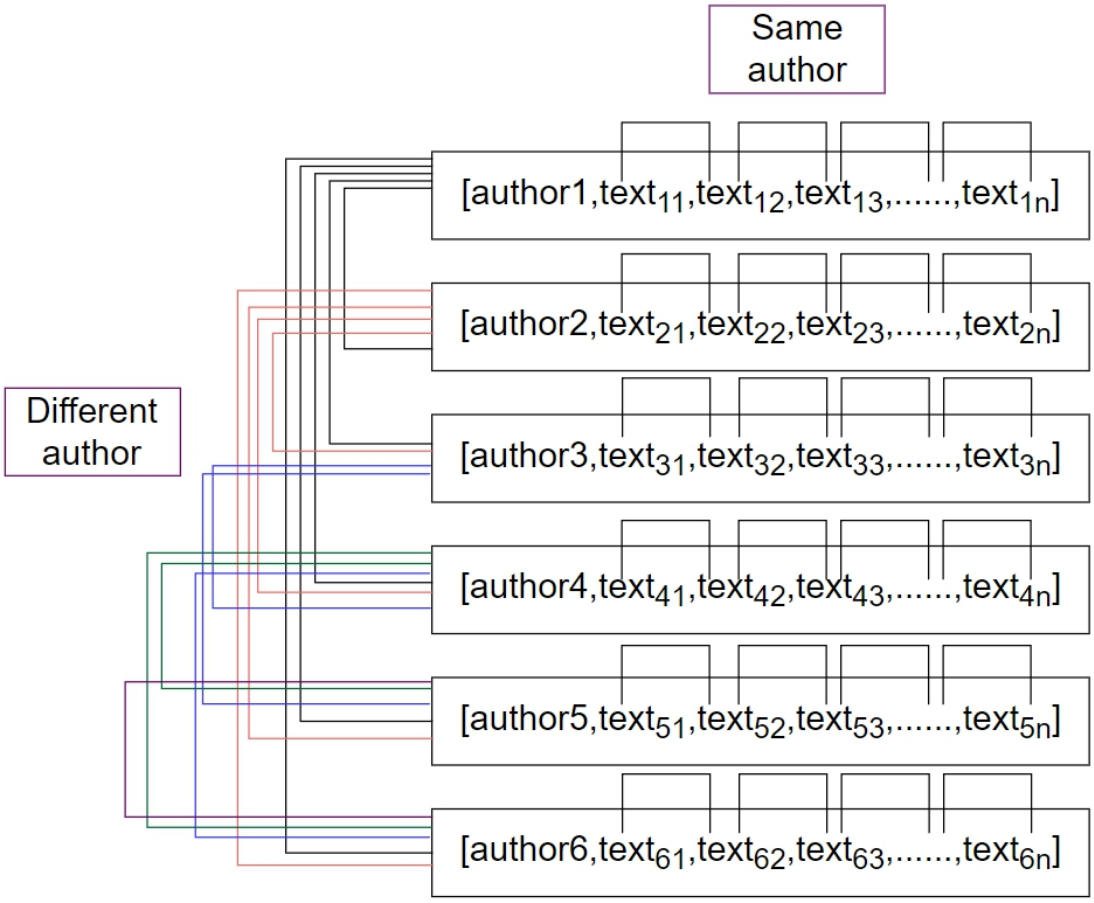

Figure 1:Augmented data texts pair matching strategy

We obtain 49,112 new training samples, including 24,472 new samples with label 1 and 24,640 new samples with label 0.

Before entering the data into the model, we first remove all emojis in the text. Secondly, we eliminate the angle brackets in the text and the messages in the angle brackets.

3.2.Network Architecture

The structure of TextCNN is the same as that of CNN. The TextCNN model is a variant of the CNN model[4]. TextCNN uses one-dimensional convolution to obtain the n-gram feature representation.

We use BERT-Base, Cased:12-Layer, 768-hidden, 12-heads, 110M parameters. There are 12 Transformers in total.

Firstly we take out each layer of the CLS features of the 12 Transformer layers one by one and add them together to calculate the mean value. Secondly, we take out each layer of the Embedding vector of the 12-layer Transformer layer and send it to TextCNN one by one. After TextCNN outputs the features one after another, they will be added together to calculate their average value. Thirdly, we concatenate the averaged CLS features with the averaged TextCNN features. Finally, the concatenated result is fed into two fully connected layers to output the classified result. The Network architecture is shown in Figure 2.

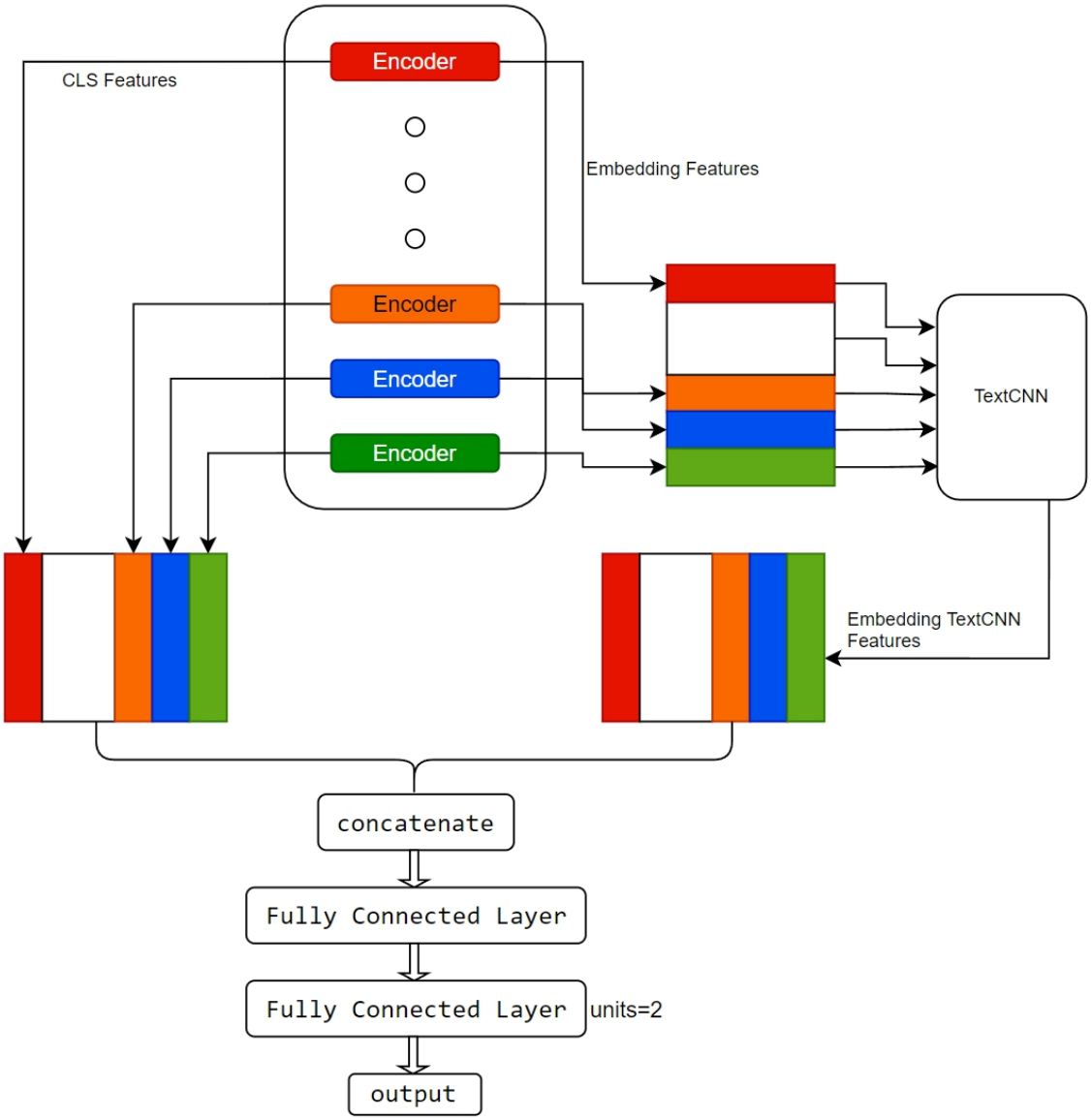

Figure 2:Network Architect diagram for our model

4. Experiments and Results

4.1.Experimental Setting

In this work, we use Keras to construct the Network Architecture. We make use of a three-layer convolution network. Firstly, we set the features map to 256 and the convolution kernels to 3, 4, and 5 , respectively. Secondly, The convolution output of each layer is fed into a GolbalAveragePolling layer. Thirdly, the first and second convolution network layer is activated by ReLU. Finally, we concatenated the pooled outputs of each layer and sent the concatenated result to a layer with a dropout rate of 0.2.

The output vector corresponding to the CLS symbol is used as the semantic representation of the entire text. The head of the shallow transformer means the shallow semantic representation, and the head of the deep transformer means the deep semantic representation. The CNN network has advantages in processing short text classification. We intend to average the heads of each layer of transformers and then input them into the CNN network to extract text features further. This model is constructed according to this idea.

We obtained 41,192 new text pairs by data reorganization as the training set. At the same time, we used 11,000 text pairs from the official data as part of the training set and 1,264 text pairs as the validation set.

The maximum length of the token we send the combination of text1 and text2 to bert is set to 128. In the last part of the network, the first fully connected layer output hidden size is set to 320, and its activation is ReLU. The second fully connected layer output hidden size is set to 2, and its activation is Softmax. We train ten epochs with the model, and its optimization is Adam with a 2e-5 learning rate.

For the final classification, we set the classification score between 0.35 and 0.65 as 0.5 as a non-judgment sample.

4.2.Results

To verify the impact of data reorganization on model performance. We split all texts of 6 authors into a validation set on the official dataset. At this time, the text on the training set contains 50 authors, and the validation set includes six authors.

Tab.le 2
Validation results on the official training dataset

Strategy	AUC	c@1	f_05_u	F1	Brier	Overall
Without Data reorganization	0.692	0.682	0.504	0.482	0.713	0.614
Data reorganization	0.791	0.753	0.613	0.556	0.776	0.698

It can be observed that with the increase in the training sample, the model's performance has improved.

Table 3 shows the final evaluation of the TIRA platform. Our model name is denoted as ye22.

Table 3
Final results on the test dataset.

Team	AUC	c@1	f_05_u	F1	Brier	Overall
ye22	0.542	0.526	0.461	0.398	0.565	0.499

5. Conclusion

In this work, we propose a method that data reorganization to augment training data and a model using Bert blend TextCNN features to solve the Authorship verification task in the PAN@CLEF 2022. This model has been experimented on the test dataset, and the result is AUC=0.542, c@1=0.526, f_05_u=0.461, F1=0.398, Brier=0.565, overall=0.499.

It can be seen from the results that our model does not perform well on the open domain authorship verification task, and in the follow-up work, we should try more efficient methods to extract the features of each text. In our work, the model with more training data leads to better performance, a breakthrough point our approach tries to improve this year.

6. Acknowledgments

This work is supported by the Social Science Foundation of Guangdong Province (No. GD20CTS02).

7. References

[1] Lin Y, Meng Y, Sun X, et al. Bertgcn: Transductive text classification by combining GCN and bert[J]. arXiv preprint arXiv:2105.05727, 2021.

[2] Peng Z, Kong L, Zhang Z, et al. Encoding text information by pre-trained model for authorship verification[C]//CLEF. 2021. J. Cohen (Ed.), Special issue: Digital Libraries, volume 39, 1996.

[3] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[4] Zhang T, You F. Research on short text classification based on textcnn[C]//Journal of Physics: Conference Series. IOP Publishing, 2021, 1757(1): 012092.

[5] Yuan X, Li Y, Xue Z, et al. Financial sentiment analysis based on pre-training and textcnn[C]//Chinese Intelligent Systems Conference. Springer, Singapore, 2020: 48-56.

[6] Y. Kim. Convolutional neural networks for sentence classification. In EMNLP, 2014.

[7] Bevendorff, Janek & Chulvi, Berta & Fersini, Elisabetta & Heini, Annina & Kestemont, Mike & Kredens, Krzysztof & Mayerl, Maximilian & Ortega-Bueno, Reyner & Pęzik, Piotr & Potthast, Martin & Rangel Pardo, Francisco & Rosso, Paolo & Stamatatos, Efstathios & Stein, Benno & Wiegmann, Matti & Wolska, Magdalena & Zangerle, Eva. (2022). Overview of PAN 2022: Authorship Verification, Profiling Irony and Stereotype Spreaders, Style Change Detection, and Trigger Detection: Extended Abstract. 10.1007/978-3-030-99739-7_42.

[8] 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.

[9] E. Stamatatos, M. Kestemont, K. Kredens, P. Pezik, A. Heini, J. Bevendorff, M. Potthast, B. Stein: Overview of the Authorship Verification Task at PAN 2022. Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2022)

[10]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.