A Contrastive Learning of Sample Pairs for Authorship Verification

Notebook for PAN at CLEF 2023

Mingcan Guo, Zhongyuan Han*, Haoyang Chen, Haoliang Qi

Foshan University, Foshan, China

Abstract

In this paper, we describe a contrastive learning method using sample pairs to compute loss for tackling the authorship verification task. Classical sample-based contrastive learning is not applicable to this task because it needs to compare multiple samples in the same batch. Our method pushes away the distance between positive sample pairs and negative sample pairs according to the cosine similarity contrast of positive and negative sample pairs so that the model has the ability to judge whether a sample pair is more similar or less similar. Evaluation results on the dataset of the PAN corpus show that the method is effective and that it could determine whether more than 50% of the sample pairs are written by the same author with an overall score greater than 0.6.

Keywords

contrastive learning, sample pairs, authorship verification, cosine similarity

1. Introduction

Text classification is a basic research direction in NLP tasks. The purpose of Authorship Verification in this direction is to judge whether two texts are written by the same person. Authorship Verification can be widely used in article duplication verification, article source finding, plagiarism detection, and other fields. In the data set of the Authorship Verification task of PAN@CLEF 2023 [1, 2], similar to last year, the organizers provide four types of text data: interview, email, essay, and speech transcription. For this task, our work builds a sentence vector model based on the naive idea of using sample pair matching labeled data, where the labeled data used are common text pair samples, and each sample is "(text1, text2, label)" format, then use the contrastive learning method of improving the loss function to complete the task. At the same time, to solve the problem of fewer training samples, we use the method of splitting and reorganizing to obtain a large amount of train data and train our model through a large number of sample pairs to improve its reasoning ability on the test set. Finally, we submit our run on TIRA.io [3].

2. Datasets

In the organizers' dataset provided by the Authorship Verification task, a total of 8836 labeled text pair samples from 56 authors are included. The label is represented by 1 or 0, representing whether the two texts are from the same author.

This year, there are four kinds of discourse types: interview, email, essay, and speech transcription. The length of each text is between 1 and 3499, and the distribution number and average length of the two texts are shown in Table 1.

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece
EMAIL: gmc9812@163.com (M. Guo); hanzhongyuan@gmail.com (Z. Han) (*corresponding author); hoyo.chen.i@gmail.com (H. Chen); haoliang.qi@gmail.com (H. Qi)
ORCID: 0000-0002-4977-2138 (M. Guo); 0000-0001-8960-9872 (Z. Han); 0000-0003-3223-9086 (H. Chen); 0000-0003-1321-5820 (H. Qi)


In PAN@CLEF 2023, the organizers firstly focuses on (cross-discourse type) authorship verification, where both written language (i.e., essays and emails) and spoken language (i.e., interviews and speech transcriptions) are represented in the set of discourse types.

Table 1
Quantity and the average length of different text types

Type	Quantity	Average length
interview	275	478
email	450	352
essay	93	388
Speech transcription	68	409

3. Methodology

3.1. Dataset Preprocessing

The training set contains a total of 56 authors. In the data set processing part, a total of 886 unique texts are obtained after deduplication. The text list is established according to the order of these authors, and each text is matched with a positive example belonging to the positive samples from the same authors or negative samples from different authors.

Specifically, suppose the extracted text list list_all= [$text_{a1}$, $text_{a2}$, $text_{b1}$, ..., $text_{z16}$], where a, b, etc. represent different authors, 1, 2, etc. represent different texts by the same author. We recombine these texts using a strategy where the first and second texts of the same author match, the second and third texts match, and a total of $\sum_{i=1}^{m}\binom{2}{n_i}$ positive sample pairs can be generated, where m is the total number of authors, n_i represents the number of texts of the i-th author, $i \in \{1,2,3,\dots,m\}$. Then match the first author's first document with the second author's random text, and the first author's second document with the third author's random text, a total of $n_i m(m-1)$ negative sample pairs can be generated. We can finally obtain 55,000 new sample data sets, such as ($text_{a1}$, $text_{a2}$, 1), ($text_{a1}$, $text_{b1}$, 0), etc.

3.2. Network Architecture

In the traditional way, most sentence vectors are formed by summing word vectors (word vectors are usually trained by methods such as word2vec). Obviously, such a method is relatively simple and crude, and the direct summing method does not utilize the interaction information between words. Instead, there are various models based on BERT. In the BERT [4] series of pre-training models, by stacking Transformer encoders, it is possible to capture the deep bidirectional word-to-word information in a sentence and use the token vector in the output layer to represent the semantic information of the entire sentence, such as BERT-flow [5] and BERT-whitening [6], etc. Our work adopts a text-based contrastive learning method. The purpose of contrastive learning is to obtain a better representation vector of text by shortening the intra-class distance and increasing the inter-class distance. Simcse [7] proved the effectiveness of contrastive learning in the text paraphrase classification task. However, Since it does not use the smallest data enhancement method to construct positive sample pairs when calculating loss, but simply uses samples that are different from itself in a batch as negative sample pairs. At the same time, a larger batch size will lead to a decrease in SimCSE performance. For this, we use a new scheme to optimize the loss function cos.

Note that Ω_{pos} is the set of all positive sample pairs, and Ω_{neg} is the set of all negative sample pairs. For any positive sample pair (i, j) $\in \Omega_{\text{pos}}$ and negative sample pair (k, l) $\in \Omega_{\text{neg}}$, there are cos (u_i, u_j) > cos (u_k, u_l), among them, u_i, u_j, u_k, u_l represent their respective sentence vectors and the new loss is shown in formula (1) and (2), where λ is the hyperparameter of the loss function.

$$D(u_k, u_l, u_i, u_j) = \lambda\big(\cos(u_k, u_l) - \cos(u_i, u_j)\big) \quad (1)$$

$$loss = \log\left(1 + \sum_{(i,j)\in\Omega_{pos},(k,l)\in\Omega_{neg}} e^{D(u_k,u_l,u_i,u_j)}\right) \quad (2)$$

Our structure is shown in Figure 1. Our work uses BERT-Large as our pre-training model, and the pre-processed sample pairs $\{t_{a1},...,t_{b1},...,t_{c1}\}$, $\{t_{a2},...,t_{c2},...,t_{d2}\}$ are respectively sent to BERT-large for encoding to obtain the vector representation of the text, then take the hidden layers of the first layer and the last layer for average pooling to obtain sentence features $\{f_{a1},...,f_{b1},...,f_{c1}\}$, $\{f_{a2},...,f_{c2},...,f_{d2}\}$, The features at the same position constitute a sample pair, and finally compare the cosine similarity of each positive sample pair with the cosine similarity of the negative sample pair by widening the distance between the positive and negative sample pairs, the positive sample pairs are closer to "more similar" and farther away from "less similar", and the negative sample pairs are closer to "less similar" and farther away from "more similar".

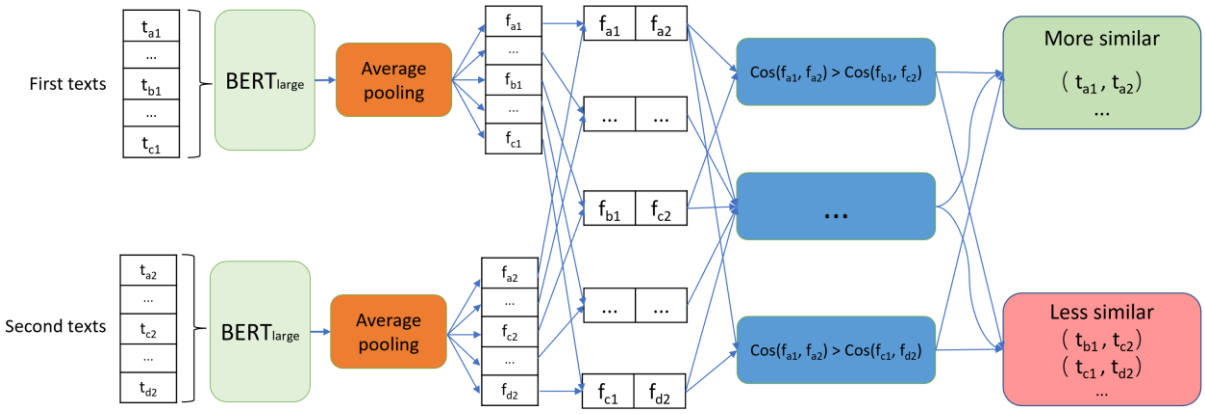

Figure 1: A contrastive learning model structure based on sample pairs, where a, b, c... represent different authors, 1,2...represent different articles belonging to the same author

4. Experiments and Results

4.1. Experimental Setting

In terms of dataset division, we preprocessed the train set and divided the train set and test set according to 7:3.

In this work, we choose BERT-Large, which has 1,024 hidden units, 24 layers and 340 million parameters. We set the batch size to 30, encoder maximum length to 512, learning rate to 2e-5, and random seed to 34. At the same time, we set the temperature coefficient λ of formula (1) to 20. We use the AdamW optimizer to update our model weights at train phase. Finally, we used an A800 for 20-epoch training.

The last layer of BERT-Large output does not select CLS, but average pools the hidden layers of the first and last layers into a new 1024-dimensional vector. In other words, the CLS embedding (of BERT-Large's output) is not used to represent the text segment pair of the input. Instead, all token embeddings except CLS and SEP are average pooled [8]. When we use BERT-Large as the encoder, we believe that the described method can obtain more comprehensive sentence features than adopting CLS embeddings.

During the prediction phase, we freeze the weights of the model to output the final result for the dataset.

4.2. Results

We obtain the organizers' two baselines for comparison, among which baseline-compressor23 is a baseline author authentication method based on text compression, which uses the partial match prediction (PPM) compression model of text1 to calculate the cross entropy of text2, and vice versa. The mean and absolute difference of the two cross-entropies is used to estimate a score in [0,1], representing the probability that the two texts were written by the same author. baseline-cngdist23 provides a simple TF-IDF weighted bag-of-character-ngrams model representation, optimized by rescaling after computing cosine similarity and projection operations so that they can act as probabilities.

In addition, we also obtained the system of najafi22, the best performer in all submissions last year, and ran our test set with reference to the parameters mentioned in the paper [9] to better evaluate our work.

To evaluate the performance of our proposed model, we used the evaluation platform provided by PAN, which includes the following metrics:

- AUC: the conventional area under the curve score.
- c@1: rewards systems that leave complicated problems unanswered [10].
- f_05_u: focus on deciding same-author cases correctly [11].
- F1: a harmonic way of combining the precision and recall of the model [12].
- Brier: Brier Score evaluates the accuracy of probabilistic predictions [13].

We input the split train data and test data into our model for training and testing, and then we use the evaluation program to evaluate the results. As shown in Table 2, our method performs best on auc, f_05_u, brier and overall.

Table 2
Performance of different methods on the split test set

Method	AUC	c@1	f_05_u	F1	Brier	overall
Ours	**0.593**	0.567	**0.581**	0.617	**0.748**	**0.621**
baseline-cngdist23	0.558	0.505	0.56	**0.671**	0.747	0.608
najafi22	0.465	**0.742**	0.495	0.662	0.55	0.583
baseline-compressor23	0.509	0.11	0.048	0.283	0.75	0.34

Ultimately, we submitted two runs, named irregular-strategist and uniform-reward. Between them, the irregular-strategist uses the 11th epoch weight of the model training (the overall performance is the best), and the uniform-reward is the 20th (the last epoch), their performance on pan23 authorship verification test is shown in Table 3. It can be seen that our best run exceeded two baselines, and the overall reached 0.614. Since the uniform-reward used the last epoch and produce overfitting problems, it only surpassed najafi22 and obtained an overall score of 0.572.

Table 3
The final performance of our submission on pan23 authorization verification test

Team	Software	AUC	c@1	f_05_u	F1	Brier	overall
pan23-cdav-1(Ours)	irregular-strategist	0.581	0.557	**0.571**	0.621	**0.742**	**0.614**
pan23-cdav-baseline	galicia22a	0.504	0.502	0.552	**0.65**	0.74	0.589
pan23-cdav-1(Ours)	uniform-reward	0.595	0.555	0.527	0.46	0.723	0.572
pan23-cdav-baseline	najafi22	**0.601**	**0.569**	0.543	0.466	0.595	0.555

5. Conclusion

This paper mainly introduces our work results on authorship verification 2023. Our work uses a sample pair contrastive learning method based on the bert-large model and improves the loss calculation function to judge whether two texts are written by the same author. Our method is effectively verified by comparing with different method or models, such as the baseline on the divided dataset. In the follow-up work, we should incorporate more effective methods to improve the performance of the

system, such as adding features in extracting the author's methods style text and compressing long text. Our method still has room for improvement.

6. Acknowledgements

This work is supported by the National Social Science Foundation of China (No. 22BTQ101).

7. References

[1] E. Stamatatos, K. Kredens, P. Pezik, Overview of the Authorship Verification Task at PAN 2023, in: CLEF 2023 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2023.

[2] J. Bevendorff, I. Borrego-Obrador, M. Chinea-Ríos, Overview of PAN 2023: Authorship Verification, Multi-Author Writing Style Analysis, Profiling Cryptocurrency Influencers, and Trigger Detection, in: Experimental IR Meets Multi-linguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), Lecture Notes in Computer Science, Springer, 2023.

[3] M. Fröbe, M. Wiegmann, N. Kolyada, Continuous Integration for Reproducible Shared Tasks with TIRA.io, in: Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2023, pp. 236–241.

[4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, Proceedings of NAACL-HLT 2019, pp. 4171–4186.

[5] B. Li, H. Zhou, J. He, On the sentence embeddings from pre-trained language models, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 9119–9130.

[6] J. Su, J. Cao, W. Liu, Y. Ou, Whitening sentence representations for better semantics and faster retrieval, arXiv preprint arXiv:2103.15316 (2021).

[7] T. Gao, X. Yao, D. Chen, Simcse: Simple contrastive learning of sentence embeddings, in: 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Association for Computational Linguistics (ACL), 2021, pp. 6894–6910.

[8] M. Huang, L. Kong, Z. Peng, Authorship verification based on fully interacted text segments, CLEF, 2022.

[9] M. Najafi, E. Tavan, Text-to-text transformer in authorship verification via stylistic and semantical analysis, in: Proceedings of the CLEF, 2022.

[10] A. Peñas, A. Rodrigo, A simple measure to assess non-response (2011).

[11] J. Bevendorff, B. Stein, M. Hagen, M. Potthast, Generalizing unmasking for short texts, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 654–659.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, Scikit-learn: Machine learning in python, the Journal of machine Learning research 12 (2011) 2825–2830.

[13] G. W. Brier, Verification of forecasts expressed in terms of probability, Monthly weather review 78 (1950) 1–3.