A Writing Style Embedding Based on Contrastive Learning for Multi-Author Writing Style Analysis

Notebook for PAN at CLEF 2023

Haoyang Chen, Zhongyuan Han*, Zengyao Li, Yong Han

Foshan University, Foshan, China

Abstract

Writing style change detection aims to identify the text position where the author switches in multi-author documents for further author identification. This paper introduces our experiment on the shared task of PAN 23. We apply the method of comparative learning in the analysis of writing style and optimize the sentence segment embedding output by the encoder of the pre-training model so that the encoder can obtain more similar vectors in space when processing sentences with similar styles, and expand the distance between the embedding representation of paragraphs with different styles. We use the optimized encoder to generate sentence embeddings by analyzing the tag data combined with paragraph sample pairs and classifying them through a full connection layer. Through experiments, we obtained F1-scores of 0.9145, 0.8203, and 0.6755 on Task 1, Task 2, and Task 3 of the official test set, respectively.

Keywords

Style Change Detection, Contrastive Learning, Sentence Representation

1. Introduction

The writing style analysis task aims to identify the position where the author's identity changes in a given multi-author document. By analyzing the author's writing style, it can help identify authorship, verify whether the document has been tampered with, whether the article is suspected of plagiarism, etc. In recent years, PAN has organized a series of tasks to detect writing style changes in the text, and conducted writing style analysis in such sub-areas as the number of authors, paragraphs with style changes, and sentence level style change detection. In PAN 2023 [1, 2], special attention was paid to the analysis of writing style under the condition of limited topic diversity, which made more attention to the article's writing style rather than the article's topic information as a signal of style change.

Since Google proposed the BERT [3] model in 2018, as a feature extraction tool from text to embedding, BERT has become increasingly popular in the NLP field. In a previous style change detection task, Zhang et al. [4] used pre-trained BERT, and Lin et al. [5] obtained the best results in the last year based on three BERT-like models using ensemble learning. However, the sentence representation directly derived from BERT is often constrained in a small area, showing high similarity, which is called "Collapse" [6], so it is difficult to be directly used for text semantic matching. Therefore, the concept of Contrastive Learning (CL) is proposed. By using the method of comparison in loss calculation, the distance between similar sentences in the vector space is closer, and different sentences are alienated. The goal is to learn a better semantic representation space from the samples. The critical point in CL is the construction of samples. Gao et al. [7] proposed SimCSE, an unsupervised solution to generate similar samples using dropout quickly, and took other sentences in the same batch as counterexamples. For supervised learning, they use the dataset (x_i, x_i^+, x_i^-) (where x_i is the premise, x_i^+ and x_i^- are entailment and contradiction hypotheses) to build the corresponding model, and

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece
EMAIL: hoyo.chen.i@gmail.com (H. Chen); hanzhongyuan@gmail.com (Z. Han) (*corresponding author); lzy1512192979@gmail.com (Z. Li); hanyong2005@fosu.edu.cn (Y. Han)
ORCID: 0000-0003-3223-9086 (H. Chen); 0000-0001-8960-9872 (Z. Han); 0000-0001-8472-4150 (Z. Li); 0000-0002-9416-2398 (Y. Han)

achieved good results. However, it is challenging to make triplet data on some tasks, so Su [8] proposed CoSENT（Cosine Sentence). This simpler but more powerful supervised CL solution enables the encoder to learn better sentence embedding representation by comparing the cosine distance between sample pairs. In this paper, CoSENT will be used to finetune an encoder to output a closer embedding in the same author's article and calculate the final label through an MLP.

2. Task and Datasets

PAN 23 provides tasks with three difficulty levels. It is necessary to find the position of all writing style changes at the paragraph level on a given text (i.e., for two consecutive paragraphs, evaluate whether there are style changes). The difficult difference of tasks lies in the diversity of document topics:

- **Easy:** The paragraphs of the document cover a variety of topics, allowing the use of topic information to detect whether the author's identity has changed.
- **Medium:** The theme in the document changes little (but still exists), forcing more attention to the style to effectively solve the detection task.
- **Hard:** All paragraphs in the document are on the same topic.

The dataset of this task provided by PAN 23 comes from Reddit and provides the user posts and their replies of each sub-section. Each dataset is divided into three parts: training set and verification set including ground truth data, and test set without ground truth data for evaluation. Table 1 provides statistics on the number of original datasets.

Table 1
Statistics of the original dataset

Datasets	Dataset 1		Dataset 2		Dataset 3	
	#documents	#para.	#documents	#para.	#documents	#para.
Training set	4200	17104	4200	32416	4200	23313
Validation set	900	3730	900	7942	900	5012

All documents are provided in English and may contain any number of style changes. However, style changes may only occur between paragraphs (i.e., a single paragraph is always authored by a single author and does not contain style changes). Each input problem is referenced by an ID (i.e., the document that detects style changes), which is then used to identify the solution submitted for the input problem. The ground truth data includes the number of authors and the binary labels of each pair of consecutive paragraphs (1 for style changes, otherwise 0), but does not provide specific paragraph author information.

3. Methodology

In this paper, we use CoSENT as a comparative learning method to train an encoder, to make the sentence embedding encoded by the paragraphs of the same author closer in space, while the sentence embedding of the paragraphs written by different authors is farther. After the encoder training, we connect a full connection layer classifier to generate the corresponding labels.

3.1. Data Pre-processing

First, the dataset needs to be converted into positive or negative instances of paragraph pairs $(P_i, P_j, label)$ to continue the subsequent comparative learning training. In the official dataset, only the binary labels and the number of authors of each pair of consecutive paragraphs are provided, which means that we cannot directly determine the author information of each paragraph. Therefore, if we directly convert the samples based on the official dataset, we will get a training set with fewer samples. If all the two passages with unchanged authorship are considered as positive instances and the other passages as negative instances, the following situation may occur: Passages that are far away but by the

same author, are incorrectly marked as negative instances, which will confuse the model. (In particular, when the number of authors is equal to the number of style changes in the document + 1, the author information of each paragraph can be uniquely determined, which can be proved by the pigeonhole principle in combinatorics, and we call it author information transparency.)

In this case, to avoid the above-mentioned problems and to make better use of the available data, we propose a method to generate positive and negative instances:

- First, divide the paragraphs whose style has not changed into the same group, based on the labels.
- If the number of paragraphs in a group is greater than one, combine each of them in two to obtain a positive instance.
- Two-by-two combinations of negative instances between two adjacent but different groups.

In this way, we can obtain a large number of high-quality paragraph pairs and ensure that the resulting positive and negative instances are always correct, although some possible negative instances may be lost. We only apply the above strategy to the training set. Only positive and negative instances pairs transformed from the original labels are used for the validation set.

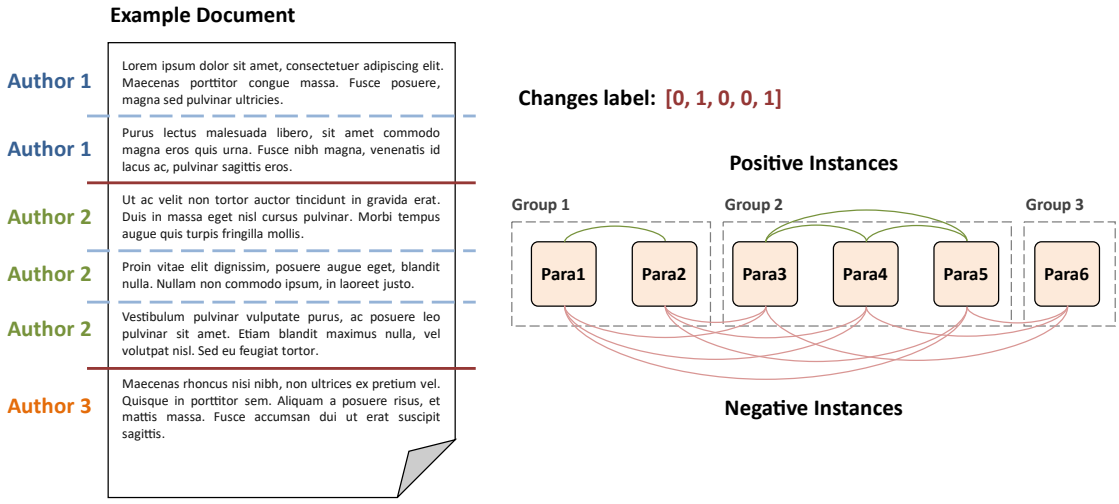


Figure 1: Dataset processing. Positive and negative instances are generated based on the division of the group.

Table 2 shows the number of paragraph pairs after dataset processing.

Table 2
Statistics of the processed dataset

Datasets	Dataset 1		Dataset 2		Dataset 3	
	#pos.	#neg.	#pos.	#neg.	#pos.	#neg.
Training set	2459	13304	46029	39125	15712	24703
Validation set	377	2451	4013	3029	2159	1953

3.2. Encoder Training

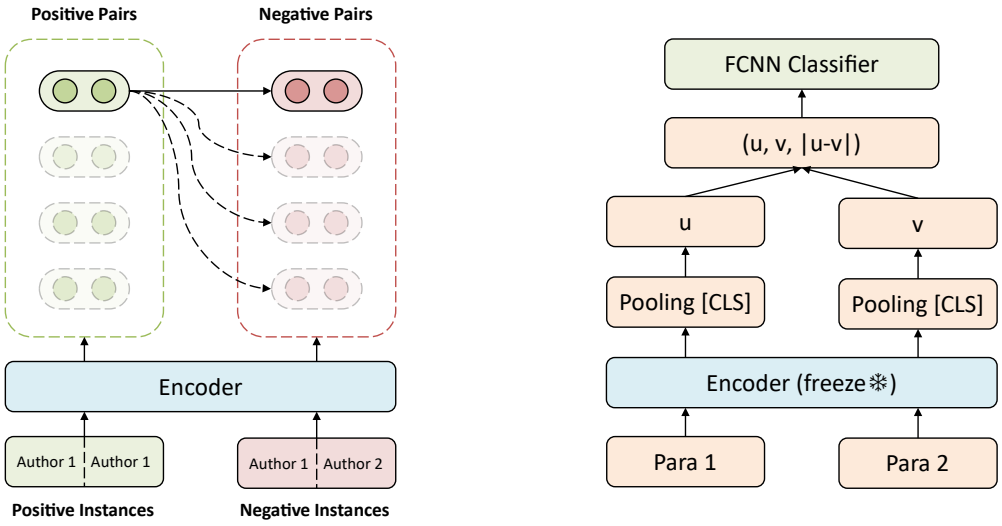
In encoder training, the positive and negative instances pairs in a batch will be sent to the encoder for encoding, and the similarity between the instances pairs will be calculated by cosine distance. We hope that the similarity of positive instances pairs is greater than that of negative instances, that is, for any positive instances pair $(i, j) \in \Omega_{pos}$ and negative instances pair $(k, l) \in \Omega_{neg}$, there are:

$$\cos(u_i, u_j) > \cos(u_k, u_l) \tag{1}$$

Where u_x represents the embedding representation of the paragraph x. The work of Su et al. [8, 9] and Sun et al. [10] suggests an effective solution to such problems, so we get the equation:

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

Where $\lambda > 0$ is a hyperparameter, which is taken as 20 in this experiment. The above equation is used to optimize the encoder, and the cosine distance of the encoder output instances is evaluated for correlation with the labels using the spearman metric, which assesses how well the relationship between two variables can be described using a monotonic function. Figure 2 depicts the overall framework of the model.



(a) Contrastive learning on encoder (b) Model structure

Figure 2: Model description. The figure on the left describes the encoder training conducted by CoSENT base on contrastive learning, and the figure on the right shows the model's overall structure.

3.3. Classifier Training

After completing the encoder training, we freeze the parameters of the encoder, then encode the instances of the paragraph pairs to be predicted, take out the last layer of the model's [CLS] vector as the embedding representation of the paragraph (u, v), subtract the two matrices and take the absolute value, and splice them with the original one into a feature matrix $(u, v, |u - v|)$, and feed it into a tanh-activated linear layer for classification. The prediction results are optimized using cross-entropy loss and evaluated by F1-scores.

4. Experiments

In the actual experiment[1], we choose the $\text{DeBERTa}_{\text{BASE}}$ [11] as our pretrain encoder model. Our hyperparameters are set as follows: For the encoder, the batch size is set to 24, the maximum sequence length is 512, and the excess will be truncated. The initial learning rate is set to 1e-5, and trained in 20 epochs; For classifiers, the batch size is set to 64, the initial learning rate is set to 5e-5, and trained in 10 epochs. AdamW was used to optimize each training, and a warmup rate of 0.1 was set.

The spearman score of the encoder model and the metrics finally obtained by the classifier can be found in Table 3.

[1] Our source code is available at https://github.com/icyray/CL-MAWSA.

Table 3
Metrics on the validation set

Datasets	Encoder	Classifier
	spearman	F1-scores
Dataset 1	0.5648	0.9054
Dataset 2	0.6941	0.8177
Dataset 3	0.4715	0.7038

5. Results

We finally submitted the model to TIRA [12] to run and obtain the final metrics of the model. Table 3 provides the scores obtained by our model in the official test set.

Table 3
Metric on the test set

Tasks	Task 1	Task 2	Task 3
F1-scores on Test set	0.9145	0.8203	0.6755

6. Conclusion

This paper briefly describes the results of our team's work on the PAN 2023 shared task. We used a CoSENT-based contrastive learning method to finetune the encoder and a linear classifier to obtain the final results. The experimental results show that contrastive learning has promising applications in this kind of tasks.

7. Acknowledgements

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

8. References

[1] J. Bevendorff, I. Borrego-Obrador, M. Chinea-Ríos, et al., 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.

[2] E. Zangerle, M. Mayerl, M. Potthast, et al., Overview of the Multi-Author Writing Style Analysis Task at PAN 2023, in: Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum, CEUR-WS, 2023.

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

[4] Z. Zhang, Z. Han, L. Kong, et al., Style Change Detection Based On Writing Style Similarity—Notebook for PAN at CLEF 2021, in: G. Faggioli, N. Ferro, A. Joly, M. Maistro, F. Piroi (Eds.), CLEF 2021 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2021.

[5] T.-M. Lin, C.-Y. Chen, Y.-W. Tzeng, et al., Ensemble Pre-trained Transformer Models for Writing Style Change Detection, in: CLEF 2022 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2022.

[6] Y. Yan, R. Li, S. Wang, et al., Consert: A contrastive framework for self-supervised sentence representation transfer, in: Proceedings of the 59th Annual Meeting of the Association for

Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 5065-5075.
[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] J. Su, CoSENT (I): A more efficient sentence embedding scheme than Sentence-BERT, 2022. URL: https://spaces.ac.cn/archives/8847
[9] J. Su, M. Zhu, A. Murtadha, et al., Zlpr: A novel loss for multi-label classification, arXiv preprint arXiv:2208.02955, 2022.
[10] Y. Sun, C. Cheng, Y. Zhang, et al., Circle loss: A unified perspective of pair similarity optimization, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, 2020, pp. 6397-6406.
[11] P. He, X. Liu, J. Gao, et al., Deberta: Decoding-enhanced bert with disentangled attention, in: International Conference on Learning Representations, 2020.
[12] M. Fröbe, M. Wiegmann, N. Kolyada, et al., 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.