A Verifying Generative Text Authorship Model With Regularized Dropout

Notebook for the PAN Lab at CLEF 2024

Zijie Lin[1], Zhongyuan Han[1,*], Leilei Kong[1], Miaoling Chen[1], Shuyi Zhang[1], Jiangao Peng[1] and Kaiyin Sun[2]

[1] *Foshan University, Foshan, China*
[2] *Foshan Huaying School, Foshan, China*

Abstract

Generative AI authorship verification aims to identify the text authored by a human within a given pair of texts. This paper presents our method for the PAN 2024 Generative AI Authorship Authentication Task. We framed this task as a binary classification problem for individual texts. Initially, we utilized data augmentation techniques to balance the originally imbalanced dataset and trained the model on single texts. Additionally, we employed the Regularized Dropout method to optimize model training further. For a given pair of texts, the model processed each text individually for inference. Finally, a fully connected layer was used for classification, selecting the text with the higher human-authorship score as the answer. Our method achieved a mean score of 0.99 on the official test set.

Keywords

PAN 2024, Generative AI Authorship Verification, Data Augmentation, Regularized Dropout

1. Introduction

Generative AI authorship verification aims to identify the text in which the author is a human, given a text pair consisting of a large model and a human-generated text. In recent years, with the innovation caused by large-scale language models such as ChatGPT [1] in assisted writing, people have increasingly relied on AI for content creation. This trend is also accompanied by several challenges and problems, such as students submitting AI-generated assignments [2] and using AI to write false articles, etc.. By verifying the identity of the author, the above-mentioned adverse phenomena can be effectively curbed. This paper describes our method to the generative AI authorship verification task [3, 4, 5] at PAN 2024. This task requires us to identify the human-authored text within a given text pair, using a limited and unbalanced human-to-machine dataset.

In this study, we framed the task as a binary classification problem for individual texts. Our method used public datasets to augment the original dataset, increasing the quantity of both human-generated and machine-generated texts to maintain equal proportions. This method of data augmentation addressed the imbalance between human and machine authors in the dataset. Furthermore, we incorporated the R-Drop [6] technique during training to enhance model robustness. During inference, the model processed each text pair as individual texts for prediction, selecting the text with the higher human authorship score as the answer.

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

lamlovezz113@gmail.com (Z. Lin); hanzhongyuan@gmail.com (Z. Han); kongleilei@fosu.edu.cn (L. Kong); 271473791@qq.com (M. Chen); shuyipro@foxmail.com (S. Zhang); wyd1n910@gmail.com (J. Peng); sunkaiyin123@163.com (K. Sun)

0009-0009-1492-809X (Z. Lin); 0000-0001-8960-9872 (Z. Han); 0000-0002-4636-3507 (L. Kong); 0009-0007-4184-4487 (M. Chen); 0009-0006-7589-8187 (S. Zhang); 0009-0006-3780-5023 (J. Peng)

2. Related Work

Due to the rapid development of large language models (LLMs), their text generation capabilities have reached a level comparable to human writing [7] . Developing effective methods to verify the authorship of generated texts is crucial for mitigating the misuse of LLMs and reducing the harmful impact of their content. In recent years, numerous studies have focused on machine text detection. For instance, Hans [8] proposed a method called Binoculars, which compares the scores of two related language models to determine whether a text is human-generated or machine-generated. Bao [9] introduced "Fast-DetectGPT," a zero-shot detection method for machine-generated text that leverages conditional probability curvature. Although these methods do not require training data and rely solely on analyzing specific textual features for detection, they may be ineffective when the characteristics distinguishing human and machine-generated texts are not prominent. Therefore, we adopted the R-Drop method to ensure consistency in the distribution of samples across different categories. The core idea of the R-Drop method is to regularize the consistency between the outputs of two different sub-models generated through dropout, thereby enhancing the model's generalization ability and robustness. This method constrains the results of two forward passes obtained by applying dropout to the same input data, ensuring they remain consistent.

3. Method

This section explains how to incorporate R-Drop to optimize our model during the training process. We use the pre-trained language model Bert [10] for training. We consider this task a binary classification problem for single text samples, thus employing the binary cross-entropy loss function as the foundation. On top of this, we incorporate the R-drop method to construct the final loss function. This final loss function is then used to train the model. The final loss function is expressed as follows:

$$\mathcal{L} = (\mathcal{L}_{BCE}(p^1, \mathrm{y}) + \mathcal{L}_{BCE}(p^2, \mathrm{y})) + \alpha(\mathcal{KL}(p^1 \parallel p^2) + \mathcal{KL}(p^2 \parallel p^1)) \tag{1}$$

Where α is a hyperparameter that controls the contribution of the KL divergence in the total loss. In this way, we consider the model's prediction accuracy and enhance the consistency of the model's results from different forward passes, thereby improving the model's stability and robustness.The specific steps for creating the loss function are as follows:

First, we input the data through the network and apply dropout to obtain two different forward propagation results p^1 and p^2. Then, we calculate the binary cross-entropy loss $\mathcal{L}_{BCE}$ for these two results. The formula for binary cross-entropy loss $\mathcal{L}_{BCE}$ is as follows:

$$\mathcal{L}_{BCE}(\mathrm{p}, \mathrm{y}) = -\sum_i [y_i \log p_i + (1 - y_i) \log(1 - p_i)] \tag{2}$$

where p is the predicted probability distribution of the model, and y is the actual label distribution. Binary cross-entropy loss measures the inconsistency between the actual labels and the predicted distribution and is a common loss function for binary classification problems.

Next, we calculate the Kullback-Leibler (KL) divergence between the two results p^1 and p^2; the formula is:

$$\mathcal{KL}(p^1 \parallel p^2) = \sum_i p_i^1 \log \frac{p_i^1}{p_i^2} \tag{3}$$

Finally, the above KL divergence is added as a regularization term to the loss function. The final loss function includes the weighted sum of binary cross-entropy loss and KL divergence loss. The application of R-drop in the training process is shown in Figure 1.

We selected the BERT model as the baseline model. We trained BERT using the training data that will be mentioned below and optimized the model using R-Drop. During the inference phase, we first split the input text pair into two separate texts. Each text is then individually fed into the BERT model for classification prediction. Finally, we select the text with the higher probability of being human-generated as the final answer.

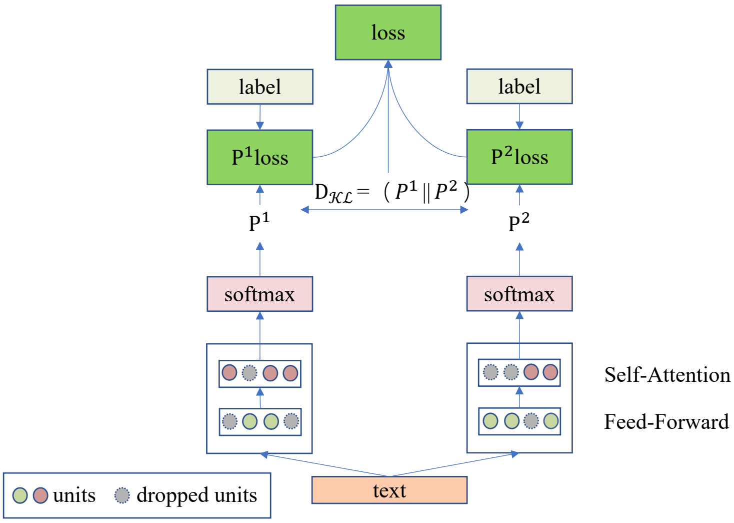

Figure 1: The figure shows that an input text passes through the same model's transformer block twice and obtains two distributions, P^1 and P^2. The KL divergence between P^1 and P^2 is then calculated. Additionally, $P^1 loss$ and $P^2 loss$ represent the binary cross-entropy losses between P^1 and the label, P^2 and label, respectively

4. Experiment

4.1. Data Preprocessing

In this task, we utilized two datasets. The first dataset is the guiding dataset provided by the organizers for the Generative AI Authorship Verification task, known as pan24-generative-authorship-news. The second dataset is sourced from the Kaggle platform, named DAIGT-V4-TRAIN-DATASET[1](hereinafter referred to as DAIGT-V4). The guiding dataset encompasses various genuine and fabricated news articles from American headlines in 2021. It comprises 14 JSONL files, with one containing text generated by human authors and the remaining 13 files containing text generated by different machine authors. The DAIGT-V4 comprises a collection of CSV files containing text generated by one human author and 11 machine authors, covering topics such as mobile phones and automobiles, with 27370 texts generated by humans and 46203 by machines. The minimum, maximum, and average lengths of texts in both pan24-generative-authorship-news and DAIGT-V4 are presented in Table 1.

Table 1
Minimum, average, and maximum length of text in different datasets

Dataset	Minimum length	Average length	Maximum length
pan24-generative-authorship-news	2	428	1389
DAIGT-V4	2	390	1671

Due to the proportion of human authors to machine authors being 1:13 in the dataset provided by the organizers, namely "pan24-generative-authorship-news," to expand the data volume and balance the ratio between human authors and machine authors, we utilized the DAIGT-V4 dataset to augment the original data. The preprocessing of the data involved extracting 1000 texts generated by human

[1] You can find this dataset at https://www.kaggle.com/datasets/thedrcat/daigt-v4-train-dataset.

authors from the pan24-generative-authorship-news dataset while retaining their respective topics. We randomly selected authors based on the same topics for the machine-generated texts. Subsequently, we extracted 20000 texts generated by both human and machine authors from the DAIGT-V4 dataset in a 1:1 ratio. We then combined these two sets of data and divided them into training and test sets at a ratio of 9:1. In the training set, a label of 1 denotes texts generated by human authors; In contrast, a label of 0 denotes texts generated by machine authors. All text will be truncated according to the maximum input length of the model.

4.2. Experimental setting

We conducted the entire experiment using the Pytorch framework. The optimizer used was the Adam optimizer. During training, the loss function was a weighted sum of binary cross-entropy loss and KL divergence, with a weight of 4 for the KL divergence. Dropout was set to 0.3, the maximum text length was 512, the batch size was 32, the learning rate was 3e-5, and the number of epochs was 10. The composition of the dataset used in the experiment is shown in Table 2.

Table 2
Composition and quantity of dataset

	DAIGT-V4	pan24-generative-authorship-news	Total
train	36000	1800	37800
test	4000	200	4200

After dividing the dataset, we send a single text to the model for training. We used the same indicators as the official PAN 2024 to evaluate our model and took the mean value as the final selection criterion for the model. We obtained the best model in the second epoch.

4.3. Other method

We also employed an ensemble learning approach to complete this task. In addition to the previously mentioned dataset, we expanded our dataset using the SemEval subTask A dataset [11] . We utilized three pre-trained language models: Bert-base-uncased, Roberta-base-uncased [12] , and Deberta-base-uncased [13] . The training process was mainly similar to the method described above. During the inference phase, we split each text pair into two separate texts and input them into the three models. Each model predicts the text separately and obtains two scores; we choose the average score as the final score for a single text and select the one with the higher score as the human-generated text.

4.4. Results

This subsection introduces the experimental results. Our team, Team lam in Table 3, submitted two systems: system $blistering{-}moss$ and system $acute{-}wireframe$. Table 3 shows an overview of the accuracy of our method and baseline methods in detecting whether humans write text in PAN 2024 (Voight-Kampff Generative AI Authorship Verification) Task 4. Among them, system $blistering{-}moss$ is our primary method, and the system $acute{-}wireframe$ is briefly introduced in Section 4.3. Compared to baseline methods, our methods demonstrate significant improvements across most metrics. For instance, the system $blistering{-}moss$ achieves an $ROC - AUC$ of 0.989, markedly higher than the highest value of 0.972 attained by all baseline methods ($Baseline\ Binoculars$). Additionally, the system $blistering{-}moss$ achieves scores of 0.989 or higher in $Brier$, $C@1$, $F1$, $F0.5u$, and the average value, indicating exceptionally high classification performance.

Although the system $acute{-}wireframe$ slightly lags behind the the system $blistering{-}moss$, it maintains all metrics around 0.865, still surpassing most baseline methods. Notably, it performs comparably to the $Baseline\ Fast{-}DetectGPT(Mistral)$ method in the $F1$ and $F0.5u$ metrics (both 0.883).

Table 3
Overview of the accuracy in detecting if a text is written by an human in task 4 on PAN 2024 (Voight-Kampff Generative AI Authorship Verification). We report ROC-AUC, Brier, C@1, F_1, $F_{0.5u}$ and their mean.

Team	System	ROC-AUC	Brier	C@1	F_1	$F_{0.5u}$	Mean
lam	blistering-moss	0.989	**0.989**	**0.989**	**0.989**	**0.99**	**0.990**
lam	acute-wireframe	0.865	0.865	0.865	0.866	0.865	0.866
Baseline Binoculars	-	0.972	0.957	0.966	0.964	0.965	0.965
Baseline Fast-DetectGPT (Mistral)	-	0.876	0.8	0.886	0.883	0.883	0.866
Baseline PPMd	-	0.795	0.798	0.754	0.753	0.749	0.77
Baseline Unmasking	-	0.697	0.774	0.691	0.658	0.666	0.697
Baseline Fast-DetectGPT	-	0.668	0.776	0.695	0.69	0.691	0.704
95-th quantile	-	0.994	0.987	0.989	0.989	0.989	0.990
75-th quantile	-	0.969	0.925	0.950	0.933	0.939	0.941
Median	-	0.909	0.890	0.887	0.871	0.867	0.889
25-th quantile	-	0.701	0.768	0.683	0.657	0.670	0.689
Min	-	0.131	0.265	0.005	0.006	0.007	0.224

Table 4 presents the average accuracy across nine test set variations. The system $blistering-moss$ demonstrates outstanding performance across all metrics: the minimum value is 0.764, indicating high accuracy even under the worst conditions; the maximum value is 0.996, indicating near-perfect accuracy under optimal conditions. The system $acute-wireframe$ also performs exceptionally well across various metrics: the minimum value is 0.015, an outlier possibly due to extreme conditions in certain test sets; the maximum value is 1.000, demonstrating perfect accuracy under the best conditions. It is worth noting that the system $acute-wireframe$ outperforms system $blistering-moss$ on all metrics except for the two German datasets.

Our methods exhibit superior performance on most metrics compared to other baseline methods. Specifically, our method significantly surpasses others in key metrics such as the median, 75th percentile, and maximum value, highlighting its robustness and efficiency.

Table 4
Overview of the mean accuracy over 9 variants of the test set. We report the minumum, median, the maximum, the 25-th, and the 75-th quantile, of the mean per the 9 datasets.

Team	System	Minimum	25-th Quantile	Median	75-th Quantile	Max
lam	blistering-moss	0.764	0.832	0.961	0.989	0.996
lam	acute-wireframe	0.015	0.843	0.866	**0.997**	**1.000**
Baseline Binoculars	-	0.342	0.818	0.844	0.965	0.996
Baseline PPMd	-	0.270	0.546	0.750	0.770	0.863
Baseline Unmasking	-	0.250	0.662	0.696	0.697	0.762
Baseline Fast-DetectGPT	-	0.159	0.579	0.704	0.719	0.982
95-th quantile	-	0.863	0.971	0.978	0.990	1.000
75-th quantile	-	0.758	0.865	0.933	0.959	0.991
Median	-	0.605	0.645	0.875	0.889	0.936
25-th quantile	-	0.353	0.496	0.658	0.675	0.711
Min	-	0.015	0.038	0.231	0.244	0.252

5. Conclusion

To solve the task of generative artificial intelligence author authentication proposed by PAN 2024, we propose two methods in this article. One is to use data augmentation and R-Drop to train the BERT model. The other is to use the Ensemble learning voting method for author verification.

The method of combining data augmentation with R-Drop yielded promising results. Despite the integrated model's overall performance potentially being inferior to the former, it demonstrated superior effectiveness on certain test data subsets.

Acknowledgments

This work is supported by the Social Science Foundation of Guangdong Province, China (No.GD24CZY02)

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., Gpt-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[2] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, C. Finn, Detectgpt: Zero-shot machine-generated text detection using probability curvature, in: International Conference on Machine Learning, PMLR, 2023, pp. 24950–24962.

[3] A. A. Ayele, N. Babakov, J. Bevendorff, X. B. Casals, B. Chulvi, D. Dementieva, A. Elnagar, D. Freitag, M. Fröbe, D. Korenčić, M. Mayerl, D. Moskovskiy, A. Mukherjee, A. Panchenko, M. Potthast, F. Rangel, N. Rizwan, P. Rosso, F. Schneider, A. Smirnova, E. Stamatatos, E. Stakovskii, B. Stein, M. Taulé, D. Ustalov, X. Wang, M. Wiegmann, S. M. Yimam, E. Zangerle, Overview of PAN 2024: Multi-Author Writing Style Analysis, Multilingual Text Detoxification, Oppositional Thinking Analysis, and Generative AI Authorship Verification, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. M. D. Nunzio, P. Galuščáková, A. G. S. de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2024.

[4] J. Bevendorff, M. Wiegmann, J. Karlgren, L. Dürlich, E. Gogoulou, A. Talman, E. Stamatatos, M. Potthast, B. Stein, Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2024, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. S. de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[5] M. Fröbe, M. Wiegmann, N. Kolyada, B. Grahm, T. Elstner, F. Loebe, M. Hagen, B. Stein, M. Potthast, Continuous Integration for Reproducible Shared Tasks with TIRA.io, in: J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), 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. doi:10.1007/978-3-031-28241-6_20.

[6] L. Wu, J. Li, Y. Wang, Q. Meng, T. Qin, W. Chen, M. Zhang, T.-Y. Liu, et al., R-drop: Regularized dropout for neural networks, Advances in Neural Information Processing Systems 34 (2021) 10890–10905.

[7] J. Wu, S. Yang, R. Zhan, Y. Yuan, D. Wong, L. Chao, A survey on llm-gernerated text detection: Necessity, methods, and future directions (2023).

[8] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting llms with binoculars: Zero-shot detection of machine-generated text, 2024. arXiv:2401.12070.

[9] G. Bao, Y. Zhao, Z. Teng, L. Yang, Y. Zhang, Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature, in: The Twelfth International Conference on Learning Representations, 2023.

[10] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. arXiv:1810.04805.

[11] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. M. Afzal, T. Mahmoud, G. Puccetti, T. Arnold, et al., Semeval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection, arXiv preprint arXiv:2404.14183 (2024).

[12] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. arXiv:1907.11692.

[13] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, 2021. arXiv:2006.03654.