Generative AI Authorship Verification Of Tri-Sentence Analysis Base On The Bert Model

Notebook for the PAN Lab at CLEF 2024

Jijie Huang[1,*], Yang Chen[1], Man Luo[2] and Yonglan Li[1]

[1]Foshan University, Foshan, China
[2]Guangzhou City University of Technology, Guangzhou, China

Abstract

The task of generative AI authorship verification aims to determine if a text is written by a human or generated by AI. In this paper, we treat this task as a binary classification problem and introduce a method called Tri-Sentence Analysis (TSA). TSA captures fine-grained contextual information, enhancing the model's ability to identify the text's source. Additionally, we incorporate the MPU method to improve the model's efficiency and differentiation for short texts.Finally, we integrated these methods into a pre-trained BERT model. On the test set, our performance metrics for the Minimum, 25-th Quantile, Median, 75-th Quantile, and Maximum scores are 0.883, 0.936, 0.976, 0.989, and 0.999, respectively.

Keywords

Authorship Verification, Tri-Sentence Analysis, Pre-trained Model

1. Introduction

In Natural Language Processing (NLP), Authorship Verification is a fundamental task, with generative AI authorship verification being particularly crucial. Its purpose is to distinguish between human-written texts and those generated by AI, ensuring content authenticity and reliability. With PAN's extensive research in authorship verification [1, 2], the PAN 2024 Generative AI Authorship Verification [3, 4] task aims to addresses this challenge. This task is organized in collaboration with the Voight-Kampff task of the ELOQUENT [5] laboratory, adopting a builder-breaker style. PAN participants must develop systems to differentiate between texts generated by large language models and those written by humans.

We consider the PAN 2024 generative AI authorship verification task as an AI text detection task, aimed at distinguishing whether a text is machine-generated or human-written. In AI text detection, models like Binoculars [6] and Fast-DetectGPT [7] make good predictions, but their performance is unsatisfactory for short AI-generated texts. We aim to determine if targeted processing of short texts can improve AI text detection efficiency. Therefore, we propose a deep learning-based text classification method called Tri-Sentence Analysis (TSA). This method divides the text into multiple short segments, each containing three sentences, and independently analyzes each segment. Then, it combines these analyses to make a final classification of the entire text. Specifically, we divide the texts in the official dataset into multiple short segments, each inheriting the original text's label, and extract features from them. By averaging the prediction values of these short segments, we determine if the original text is AI-generated or human-written. To improve short text classification, we employ a method called MPU [8], which enhances the efficiency of short text recognition. Finally, we use the pre-trained language model BERT [9] to complete this task and submit our model on TIRA [10].

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France
2112203035@stu.fosu.edu.cn (J. Huang); 980842599@qq.com (Y. Chen); luoman322@163.com (M. Luo); li_yonglan@163.com (Y. Li)
0000-0002-8462-3310 (J. Huang); 0009-0009-2368-3565 (Y. Chen); 0009-0004-1007-9100 (M. Luo); 0009-0008-5095-3404 (Y. Li)

2. Background

The rapid development of artificial intelligence technology has led to significant advancements in large language models (LLMs) for text generation. These models can produce well-structured and grammatically correct text and are widely used in various fields, including advertising, news writing, storytelling, and code generation. However, some malicious individuals misuse LLMs for harmful purposes, such as creating credible fake news or cheating, which misleads readers and has severe negative societal impacts. Therefore, distinguishing AI-generated text from human-written text to prevent abuse has become urgent. Detection methods for text generated by large models can be divided into two main categories:

The first method is zero-shot detection, which identifies AI-generated text by directly accessing the source model that created the text. This method does not need pre-trained datasets but uses the source model's output logits or loss values to determine if the text is machine-generated. Examples include the methods by Mitchell et al. [11] and Yang et al. [12] The advantage of zero-shot detection is that it does not require large amounts of training data and can be applied directly to new text. However, its disadvantage is that it depends on the performance of the source model or a proxy model. If there is a significant difference between the proxy model and the source model, the detection effectiveness may be low.

The second method is based on deep neural network (DNN) classifiers, which detect human-written and AI-generated text through supervised training models. The advantage of this method is that it can improve detection performance through large amounts of training data. For example, the method by Guo et al. [13]. However, DNN-based classifiers have high data requirements, poor generalization ability [14], and the trained classifiers are vulnerable to backdoor attacks [15] and adversarial attacks [16].

3. System Overview

3.1. Data Preprocessing

Table 1
Overview of the datasets used in PAN 2024 Generative AI Authorship Verification task.

Dataset	Quantity	Label
human	1087	1
alpaca-7b	1087	0
bigscience-bloomz-7b1	1087	0
chavinlo-alpaca-13b	1087	0
gemini-pro	1087	0
gpt-3.5-turbo-0125	1087	0
gpt-4-turbo-preview	1087	0
meta-llama-llama-2-70b-chat-hf	1087	0
meta-llama-llama-2-7b-chat-hf	1087	0
mistralai-mistral-7b-instruct-v0.2	1087	0
mistralai-mixtral-8x7b-instruct-v0.1	1087	0
qwen-qwen1.5-72b-chat-8bit	1087	0
text-bison-002	1087	0
vicgalle-gpt2-open-instruct-v1	1087	0

The dataset for the author verification task in generative AI, provided by ELOQUENT and PAN participants, includes various types of texts such as news articles, Wikipedia summaries, and fan fiction. It covers real and fake news articles from multiple US headlines in 2021. The dataset consists of 14 JSONL files, each containing 24 topics and 1087 articles. One file is written by humans, while the other 13 files are generated by different large language models. Each file corresponds to the same row ID, indicating the content pertains to the same topic. In total, the 14 files contain 15,218 articles, as shown

in Table 1. Label 1 indicates text written by humans, while label 0 indicates text generated by large language models.

We integrated and categorized the dataset provided by PAN into a new dataset named "combine." This dataset consists of two columns: "text" and "label." The "text" column represents the content of the text, while the "label" column indicates the source of the text, with human-written texts labeled as 1 and machine-generated texts labeled as 0. The combine dataset contains a total of 15,218 articles. We used 80% of the data labeled as 0 and 1 for training and the remaining 20% for validation, resulting in 12,174 samples for training and 3,044 samples for validation.

3.2. Method

Our objective is to split long texts into multiple short texts and determine whether the original text was generated by AI or written by humans by analyzing features extracted from each short text.To achieve this, we propose a deep learning-based text classification method called Tri-Sentence Analysis (TSA). TSA works by dividing long texts into short texts, each containing three sentences, and analyzing each independently.The combined results of these analyses are used for the final classification of the entire text.This approach aims to capture fine-grained contextual information more effectively, thereby improving classification accuracy. Additionally, segmenting the text reduces the burden of long texts on the model, enhancing its stability when processing lengthy texts.

During the prediction phase, we use TSA to process new input texts in a similar manner.Each group of three sentences is treated as a short text and input into the trained BERT model. The results of each short text prediction are averaged with weights to classify the original text. During testing, if we need to determine which of two texts is closer to being human-written, we apply the same method to obtain the prediction values for each text. The text with a prediction value closer to human-written is classified as such. We also want to enhance the model's ability to classify short texts. Tian et al. [8] proposed a new loss function called MPU, which improves the recognition and differentiation of AI-generated short texts.Therefore, we modified the model's loss function to MPU [8] to improve classification accuracy. Our system architecture is shown in Figure 1.

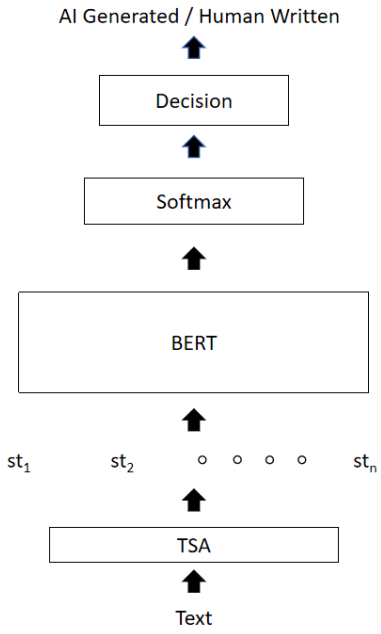

Figure 1: Model skeleton of our method

4. Experiments and Results

4.1. Experimental Setting

In this work, we selected the BERT-base-uncased model, featuring 12 layers, 768 hidden units, 12 attention heads, and 110M parameters. The maximum length of the encoder was configured to 512, with a batch size of 32. Meanwhile, we employed MPU [8] as the loss function, utilizing the Adam optimizer with a learning rate of 5e-5. Our experiments were conducted on an A800 server. The optimal performance was achieved after 13 epochs of training.

4.2. Experimental Setting

To evaluate the performance of our 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 [17].
- **F_0.5u**: focus on deciding same-author cases correctly [18].
- **F1-score**: harmonic way of combining the precision, and recall of the model [19].
- **Brier**: Brier Score evaluates the accuracy of probabilistic predictions [20].
- **Mean**: The arithmetic mean of all the metrics above.

4.3. Result

We finally submitted the model to TIRA [10] for execution to obtain the final metrics. Our model, charitable-mole_v3, performed exceptionally well in the PAN 2024 Generative AI Authorship Verification task. Table 2 presents the outstanding performance of charitable-mole_v3 across various metrics: ROC-AUC of 0.991, Brier of 0.991, C@1 of 0.991, F1 of 0.99, F0.5u of 0.989, and Mean of 0.99. Our model outperformed the official baselines across all metrics and maintained strong competitiveness in the 95th percentile among participants.

Table 3 further illustrates the average accuracy of charitable-mole_v3 across different dataset variants, particularly on the test sets of nine variants. Our model's Minimum value across all variants was 0.883, with the 25-th and 75-th Quantile at 0.936 and 0.989, respectively, a Median of 0.976, and a Maximum value of 0.999. These results significantly surpass those of all official baselines.

Compared to the quantile results of other participants, charitable-mole_v3 is close to or surpasses the models in the 95-th quantile in most metrics and exceeds the 75-th quantile models in all metrics, demonstrating strong competitiveness. This further proves the excellent performance of our model in the PAN 2024 Generative AI Authorship Verification task.

5. Conclusion

This paper details our achievements in the PAN2024 generative AI author verification task. We proposed a text classification method based on the BERT pre-trained model, called Tri-Sentence Analysis (TSA). The TSA method can capture more fine-grained contextual information, thereby improving the accuracy of text classification. It better understands the semantic relationships and consistency between sentences, enhancing the model's robustness in handling long texts. Additionally, we integrated the MPU method to improve the efficiency of distinguishing short texts. Ultimately, our method performed excellently in the PAN2024 generative AI author verification test set. The Minimum, 25-th Quantile, Median, 75-th Quantile, and Maximum values were 0.883, 0.936, 0.976, 0.989, and 0.999, respectively. In the future, we plan to further improve this method and explore its potential applications in a broader range of natural language processing tasks to achieve higher detection efficiency and wider application.

Table 2
The final performance of our submission on PAN 2024 (Voight-Kampff Generative AI Authorship Verification)

Approach	ROC-AUC	Brier	C@1	F_1	$F_{0.5u}$	Mean
charitable-mole_v3	0.991	0.991	0.991	0.99	0.989	0.99
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 3
Overview of the mean accuracy over 9 variants of the test set

Approach	Minimum	25-th Quantile	Median	75-th Quantile	Max
charitable-mole_v3	0.883	0.936	0.976	0.989	0.999
Baseline Binoculars	0.342	0.818	0.844	0.965	0.996
Baseline Fast-DetectGPT (Mistral)	0.095	0.793	0.842	0.931	0.958
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

References

[1] E. Stamatatos, M. Kestemont, K. Kredens, P. Pezik, A. Heini, J. Bevendorff, B. Stein, M. Potthast, Overview of the authorship verification task at pan 2022, in: CEUR workshop proceedings, volume 3180, CEUR-WS. org, 2022, pp. 2301–2313.

[2] J. Bevendorff, I. Borrego-Obrador, M. Chinea-Ríos, M. Franco-Salvador, M. Fröbe, A. Heini, K. Kredens, M. Mayerl, P. Pęzik, M. Potthast, et al., Overview of pan 2023: Authorship verification, multi-author writing style analysis, profiling cryptocurrency influencers, and trigger detection: Condensed lab overview, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2023, pp. 459–481.

[3] 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.

[4] 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.

[5] J. Karlgren, L. Dürlich, E. Gogoulou, L. Guillou, J. Nivre, M. Sahlgren, A. Talman, Eloquent clef shared tasks for evaluation of generative language model quality, in: European Conference on Information Retrieval, Springer, 2024, pp. 459–465.

[6] 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, arXiv preprint arXiv:2401.12070 (2024).

[7] G. Bao, Y. Zhao, Z. Teng, L. Yang, Y. Zhang, Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature, arXiv preprint arXiv:2310.05130 (2023).

[8] Y. Tian, H. Chen, X. Wang, Z. Bai, Q. Zhang, R. Li, C. Xu, Y. Wang, Multiscale positive-unlabeled detection of ai-generated texts, arXiv preprint arXiv:2305.18149 (2023).

[9] J. Lee, K. Toutanova, Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 3 (2018) 8.

[10] 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.

[11] 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.

[12] X. Yang, W. Cheng, Y. Wu, L. Petzold, W. Y. Wang, H. Chen, Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text, arXiv preprint arXiv:2305.17359 (2023).

[13] B. Guo, X. Zhang, Z. Wang, M. Jiang, J. Nie, Y. Ding, J. Yue, Y. Wu, How close is chatgpt to human experts? comparison corpus, evaluation, and detection, arXiv preprint arXiv:2301.07597 (2023).

[14] A. Uchendu, T. Le, K. Shu, D. Lee, Authorship attribution for neural text generation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 8384–8395.

[15] F. Qi, M. Li, Y. Chen, Z. Zhang, Z. Liu, Y. Wang, M. Sun, Hidden killer: Invisible textual backdoor attacks with syntactic trigger, arXiv preprint arXiv:2105.12400 (2021).

[16] X. He, X. Shen, Z. Chen, M. Backes, Y. Zhang, Mgtbench: Benchmarking machine-generated text detection, arXiv preprint arXiv:2303.14822 (2023).

[17] A. Peñas Padilla, Á. Rodrigo Yuste, A simple measure to assess non-response (2011).

[18] 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.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, the Journal of machine Learning research 12 (2011) 2825–2830.

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