# Generative AI Authorship Verification based on Contrastive Learning and Domain Adaptation

Notebook for the PAN Lab at CLEF 2024

Kaicheng Huang, Haoliang Qi* and Kai Yan

*Foshan University, Foshan, Guangdong, China*

**Abstract**

Generative AI Authorship Verification is a task that is given two pieces of text, one is a human text, and the other is a text generated by AI, and determines which of them is a human text. In this paper, we accomplish this task(Generative AI Authorship Verification) by contrastive learning, domain adaptation, and pre-trained language models. Compared with traditional machine learning methods, we use self-supervised contrastive learning and unsupervised domain adaptation methods to effectively utilize labeled source domain and unlabeled target domain data, obtain features in human texts and AI texts, and use these Features to classify the text. It can be seen from our experiments that on the validation set we constructed by ourselves, the average score of our model in ROC-AUC, Brier, F1, c@1, and F0.5U reached 0.994, and the average score in the PAN test data was the score reached 0.480.

**Keywords**

PAN 2024, Generative AI Authorship Verification, Contrastive Learning, Domain Adaptation, Pre-trained Model

## 1. Introduction

As large language models (LLMs) improve and become more widely adopted, distinguishing between human and machine-written text becomes increasingly challenging. Many classification methods exist to help with this differentiation, but the fundamental feasibility of the task is rarely questioned. Drawing on many years of experience in a related but broader field (authorship verification), we set out to answer whether this task can be solved. The goal of the PAN@CLEF 2024 generative AI author verification task [1] [2] is: given two texts, one written by a human and one written by a machine, pick out the human.

In recent years, pre-trained language models have gradually matured and can be adapted to more and more tasks, one of which is author identification. With contrastive learning and domain adaptation methods being steadily developed, in order to understand the source of the text, such as ConDA [3], a framework that use self-supervised contrastive learning and unsupervised text detection to label through labeled data sets—unlabeled text to detect whether AI generates text from different sources. In this paper, we will use the ConDA framework and the pre-trained language model RoBERTa [4] to complete the AI text recognition task and discuss its effectiveness.

## 2. Background

The birth of generative text detection is mainly due to the increasing capabilities of LLMs. Without specific training or guidance, the authenticity of the content generated by these LLMs is uncontrollable, such as generating false news, patents, etc. Content harmful to society may be misleading or politically incorrect to readers without good subjective judgment. To verify whether the text is human-written or generated by AI, combined with Generative AI Authorship Verification @ PAN, we will explore this work through different features generated by human writing and AI.

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

*Corresponding author.

teaslation@gmail.com (K. Huang); haoliang.qi@gmail.com (H. Qi); yankai@fosu.edu.cn (K. Yan)

0000-0002-3473-3355 (K. Huang); 0000-0003-1321-5820 (H. Qi); 0000-0002-4960-7108 (K. Yan)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

CEUR-WS.org/Vol-3740/paper-244.pdf

Since AI-generated texts are difficult to distinguish by manual comparison alone, many methods have been proposed in recent years for AI detection work in order to detect whether text is generated by AI. For example, from simple feature-based classifiers to fine-tuned language model detectors to distinguish whether the input text is written by humans or generated by AI [5], including detection methods specifically for AI-generated news [6]; a related research direction is author attribution (AA). Although early AA methods focused on human authors, recent studies have built models to identify specific input text generators [7]. There is also the framework ConDA designed for AI text detection based on self-supervised contrastive learning and unsupervised language adaptation used in this article.

Contrastive learning [8] focuses on comparing similar and dissimilar samples to learn data representations, creating high-quality features without explicit labels. This enhances generalization, which is crucial for transfer learning across various tasks and data distributions.

Domain adaptation [9] helps machine learning models generalize from a source domain to a different target domain, addressing performance degradation due to distribution differences. By fitting the model to the target domain's data distribution, domain adaptation improves the model's performance on new data.

Self-supervised contrastive learning leverages the inherent structural information of data to learn high-quality feature representations without labels. When combined with unsupervised domain adaptation, this approach enhances the model's ability to accurately detect AI-generated text across various generators without relying on extensive labeled data. This combination allows the model to learn robustly in both source and target domains, acquiring universal features that improve performance in the target domain.

## 3. Model Framework

In this paper, we adopt the contrastive learning domain adaptation framework and use the RoBERTa model to obtain text features. The framework consists of a Source Domain(S) and a Target Domain(T). During training, we input two texts into S and T respectively. The text input into the S is labeled, whereas the text input into the T is unlabeled. For the S part, this part mainly inputs the labeled data set. The data set can be expressed as $S = \{x_i^S, y_i^S\}$, $x_i^S$ represents the input article, and $y_i^S$ is the label of $x_i^S$, marking whether $x_i^S$ was written manually or by AI-generated. For the T part, this part mainly inputs unlabeled data sets. The data set is represented as $T = \{x_i^T\}$. The loss of T tags is mainly for S to predict in T through model adaptation. The input articles from the source ($x_i^S$) and target ($x_i^T$) domains undergo a text transformation to generate transformed samples $x_j^S$and $x_j^T$. This transformation helps in aligning the representations of source and target domain texts.

To enable the original samples and the converted samples to be input at the same time and share the weights of RoBERTa, we use a Siamese network. Through RoBERTa, use the [CLS] tag to obtain the hidden layers of the input text: $h^S_{i[\mathrm{CLS}]}$ and $h^S_{j[\mathrm{CLS}]}$, and pass these embeddings into a projection layer composed of MLP and hidden layers, and calculate the contrastive loss in their low-dimensional projection space. The contrastive loss for the source is denoted by:

$$\mathcal{L}^S_{ctr} = -\sum_{(i,j)\in b} \log \frac{\exp\left(\frac{\mathrm{sim}\left(z_i^S, z_j^S\right)}{t}\right)}{\sum_{k=1}^{2|b|} \mathbb{1}_{[k\neq i]} \exp\left(\frac{\mathrm{sim}\left(z_i^S, z_k^S\right)}{t}\right)} \tag{1}$$

where $z_i^S$ and $z_j^S$ denote the projection layer embeddings for the original and the transformed text, $t$ is the temperature, $b$ is the current mini-batch, $sim(\cdot,\cdot)$ is a similarity metric which is cosine similarity in our case. In the target, the contrastive loss is represented by $L_ctr^T$, and the equation is the same as (1).

For both the original text and the transformed text in the source domain, the CE loss is computed to train the model to classify the text instances as either human-written or AI-generated correctly. The

transformation performed on the original text preserves the semantics of the text and hence is label-preserving. In this case, we hope the classifier can also detect text with such small, semantic-preserving perturbations. This not only improves the robustness of the classifier but also increases the versatility of the detector. This binary classification task helps in learning to differentiate between the two types of text. The CE loss for the source is denoted by:

$$\mathcal{L}_{CE}^{S} = -\frac{1}{b}\sum_{i=1}^{b}\left[y_i \log p\left(y_i \mid h_{i[CLS]}^{S}\right) + (1-y_i)\log\left(1 - p\left(y_i \mid h_{i[CLS]}^{S}\right)\right)\right] \tag{2}$$

where $L_{CE}^{S}$ denotes the CE loss for the original text, $b$ denotes the batch size. The CE loss of the transformed text is represented by $L_{CE}^{S'}$, and the equation is the same as (2).

MMD [10] is utilized to align the distributions of text embeddings between the source domain (labeled data) and the target domain (unlabeled data). By minimizing the MMD, the model aims to reduce the distributional dissonance between the two domains, ensuring that the learned representations are domain-invariant. This encourages the model to learn domain-invariant features that effectively detect AI-generated text across different generators. The MMD is denoted by:

$$MMD(\mathcal{S},\mathcal{T}) = \|\frac{1}{N^S}\sum_{i=1}^{N^S}\phi\left(z_i^S\right) - \frac{1}{N^T}\sum_{i=1}^{N^T}\phi\left(z_i^S\right)\|_{\mathcal{H}} \tag{3}$$

where $S = \{x_1^S, x_2^S, x_3^S \ldots, x_N^S\}$ and $T = \{x_1^T, x_2^T, x_3^T \ldots, x_N^T\}$ are two samples drawn from the distributions $\mathcal{S}$ and $\mathcal{T}$. $\phi : \mathcal{S} \mapsto \mathcal{H}$ and H represents the RKHS space [11]. The RKHS mapping helps in aligning the feature representations of the source and target domains in a higher-dimensional space. By mapping the samples to a common RKHS, the MMD calculation aims to minimize the distributional dissonance between the domains and learn domain-invariant representations that are effective for domain adaptation.

The final training objective for our main framework is:

$$\mathcal{L} = \frac{(1-\lambda_1)}{2}\left[L_{CE}^{S} + L_{CE}^{S'}\right] + \frac{\lambda_1}{2}\left[L_{ctr}^{S} + L_{ctr}^{T}\right] + \lambda_2 MMD(S,T) \tag{4}$$

where $\lambda_1$ and $\lambda_2$ are hyper-parameters.

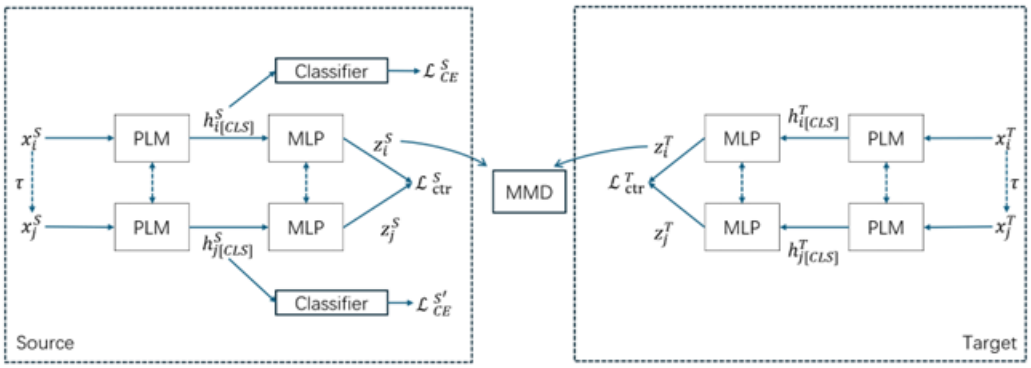


**Figure 1:** Model Framework. The PLM refers to the pre-train language model (In this paper, it means roberta-base); the PLM and MLP weights are shared across all four instances. The weights of S and T are independent of each other to adapt to the specific characteristics and distribution of each domain. The final classes are derived in the model during inference by passing the extracted features to the classification head

# 4. Experiment and Result

## 4.1. Experiment Setting

This paper chooses RoBERTa-base as an encoder with 12-layer, 768-hidden, 12-heads, and 110M parameters. The vocab size is 50,265. The maximum length of the encoder is set to 512. We used Adam optimizer with the learning rate set to 2e-5. Our experiment was conducted on an A800 server. The best performance is achieved through 10 epoch models.

## 4.2. Dataset

### 4.2.1. PAN Dataset

PAN@CLEF 2024 generative AI author verification task provides a bootstrap dataset of real and fake news articles containing multiple 2021 US news headlines. The test set which includes contributions from ELOQUENT participants, encompasses a variety of text types such as news articles, Wikipedia intro texts, and fanfiction.The bootstrap dataset contains human text and text generated by multiple large language models. There are 1,087 news topics, and each topic has human text and text generated by various Ais, such as Alpaca, ChatGPT, LLaMA, etc.

### 4.2.2. External Dataset

For comparative learning, we also added an external dataset, TT-Grover-mega [12]. This dataset is made for fake news detection. It contains text and labels. The labels are divided into human and grover_mega. Human corresponding to human text, grover_mega is text generated by the Grover generation model. This dataset has a strong correlation with the AI author recognition task. The types and number of labels of the training set, validation set, and test set of this dataset are shown in Table 1.

**Table 1**
The types and number of labels of TT-Grover-mega

| TT-Grover-mega Dataset | Human number | Grover number |
|---|---|---|
| Train | 5964 | 5507 |
| Validation | 975 | 894 |
| Test | 1915 | 1763 |

## 4.3. Data Preprocessing

To allow the dataset to be used smoothly in experiments, we split the bootstrapping data set into a training set, a verification set, and a test set. According to the human category and AI category, we split these two types of datasets into training sets, verification sets, and test sets in a ratio of 9:1:2. For the TT-Grover-mega Dataset, we retain its original quantity and modify its data format to the same JSONL format as the bootstrap dataset.

## 4.4. Data Augmentation

To expand the dataset and improve the model's generalization ability, we performed data enhancement on the bootstrap dataset and TT-Grover-mega Dataset. Through sentence segmentation, we traversed each article in the data set and extracted 10% of the words in the original article. Replace them with their synonyms, and then recombine the enhanced sentences into articles. The enhanced data is combined with the original data and their labels to generate an improved dataset.

### 4.5. Evaluation

When evaluating the model, we separate two sentences from the test set and make predictions. If the model can accurately identify the types of both texts, we output a label in the range of (0, 1) as required. If the two texts are recognized as the same category, indicating that the model is confused, we assign a label of 0.5. To evaluate the performance of our model, we used the evaluation platform provided by PAN, which includes the following metrics:

- **ROC-AUC**: the conventional area under the curve score.
- **c@1**: rewards systems that leave complicated problems unanswered [13].
- **F_0.5u**: focus on deciding same-author cases correctly [14].
- **F1-score**: harmonic way of combining the precision and recall of the model [15].
- **Brier**: Brier Score evaluates the accuracy of probabilistic predictions [16].

### 4.6. Results

Table 2 shows our experimental results.We conducted two experiments. Initially, we used TT-Grover-mega as the source and the bootstrap data set as the target. However, this approach yielded suboptimal results. Subsequently, we reversed the positions of the two samples. We hypothesize that this phenomenon can be attributed to the minimal distributional disparity between the experimental test set and the provided training set, thereby hindering our model's domain adaptation capabilities. We observed a significant improvement in our results after switching the data sets.

The first row is the bootstrap dataset as Source and the TT-Grover-mega Dataset as Target. The second row is the results when the TT-Grover-mega Dataset is used as the Source and the bootstrap dataset is used as the Target. We found that using the bootstrap dataset as the source and the TT-grover-mega dataset as the target led to a significant improvement compared to swapping the two datasets.

Table 3 demonstrates the performance of our model($\text{PAN}_{\text{Source}}$, $\text{TT}_{\text{Target}}$) evaluated on the TIRA [17] environment for PAN@CLEF 2024.

Table 4 demonstrates the final results obtained by our model in PAN@CLEF 2024

**Table 2**
The results of the test set

| Approach | ROC-AUC | Brier | C@1 | $F_1$ | $F_{0.5u}$ | Mean |
|---|---|---|---|---|---|---|
| Robert($TT_{Source}$,$PAN_{Target}$) | 0.658 | 0.778 | 1 | 0.359 | 0.41 | 0.641 |
| Robert($PAN_{Source}$,$TT_{Target}$) | 1 | 0.994 | 1 | 0.987 | 0.99 | 0.994 |

**Table 3**
Results on pan24-authorship-verification-test

| Approach | ROC-AUC | Brier | C@1 | $F_1$ | $F_{0.5u}$ | Mean |
|---|---|---|---|---|---|---|
| current-boutique | 0.951 | 0.924 | 0.922 | 0.844 | 0.868 | 0.902 |
| 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 |

**Table 4**
Final results on pan24-authorship-verification

| Approach | ROC-AUC | Brier | C@1 | $F_1$ | $F_{0.5u}$ | Mean |
|---|---|---|---|---|---|---|
| current-boutique | 0.645 | 0.798 | 0.325 | 0.307 | 0.323 | 0.480 |

## 5. Conclusion

We successfully completed the PAN@CLEF2024 generative AI authorship verification task in this benchmark task through self-supervised contrastive learning and unsupervised domain adaptation. Our results surpassed four baselines and achieved a mean score of 0.902, effectively distinguishing most human-written texts from AI-generated texts. We found that if there are more text source generators (referred to as LLMs), our method can capture the characteristics of the text more accurately and can judge more texts of unknown origin, thereby determining whether the text is manually written or generated by AI. We will take multi-language into consideration to achieve higher adaptability in the future.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.62276064).

## References

[1] 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.

[2] 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.

[3] A. Bhattacharjee, T. Kumarage, R. Moraffah, H. Liu, Conda: Contrastive domain adaptation for ai-generated text detection, in: Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Nusa Dua, Bali, 2023, pp. 598–610. URL: https://aclanthology.org/2023.ijcnlp-long.40.

[4] 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, arXiv preprint arXiv:1907.11692 (2019).

[5] D. Ippolito, D. Duckworth, C. Callison-Burch, D. Eck, Automatic detection of generated text is easiest when humans are fooled, arXiv preprint arXiv:1911.00650 (2019).

[6] T. Kumarage, J. Garland, A. Bhattacharjee, K. Trapeznikov, S. Ruston, H. Liu, Stylometric detection of ai-generated text in twitter timelines, arXiv preprint arXiv:2303.03697 (2023).

[7] S. Munir, B. Batool, Z. Shafiq, P. Srinivasan, F. Zaffar, Through the looking glass: Learning to attribute synthetic text generated by language models, in: Proceedings of the 16th Conference of

the European Chapter of the Association for Computational Linguistics: Main Volume, 2021, pp. 1811–1822.
[8] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, J. Tang, Self-supervised learning: Generative or contrastive, IEEE transactions on knowledge and data engineering 35 (2021) 857–876.
[9] M. Long, H. Zhu, J. Wang, M. I. Jordan, Unsupervised domain adaptation with residual transfer networks, Advances in neural information processing systems 29 (2016).
[10] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, A. Smola, A kernel two-sample test, The Journal of Machine Learning Research 13 (2012) 723–773.
[11] I. Steinwart, On the influence of the kernel on the consistency of support vector machines, Journal of machine learning research 2 (2001) 67–93.
[12] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, Y. Choi, Defending against neural fake news, Advances in neural information processing systems 32 (2019).
[13] A. Peñas, A. Rodrigo, A simple measure to assess non-response (2011).
[14] J. Burstein, C. Doran, T. Solorio, 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), 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.
[15] 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.
[16] G. W. Brier, Verification of forecasts expressed in terms of probability, Monthly weather review 78 (1950) 1–3.
[17] 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.