BCAV: A Generative AI Author Verification Model Based on the Integration of Bert and CNN

Notebook for PAN at CLEF 2024

Guihong Sun, Wenyin Yang* , Li Ma

Foshan University, Foshan, China

Abstract

As large language models (LLMs) continue to advance at astonishing speeds and are increasingly widely adopted, it becomes increasingly difficult for people to discern whether a given text is written by a human or a machine. Authorship verification has become a crucial and challenging task. This paper employs a text classification model that combines BERT and Convolutional Neural Networks (CNNs) in order to leverage BERT's powerful contextual understanding capabilities and CNN's efficient local feature extraction abilities to enhance text classification performance. The introduction of CNN effectively compensates for BERT's shortcomings in extracting features at the phrase level, particularly in capturing local features in the text, such as capturing n-gram features. Experimental results demonstrate that our approach outperforms baseline models significantly, with improvements of up to 6% in the ROC-AUC metric and nearly 3% in the Mean metric. We thus validate the effectiveness of this approach.

Keywords

PAN 2024, Generative AI Authorship Verification, BERT, CNN

1. Introduction

The Generative AI Authorship Verification Task @ PAN [1]is organized in collaboration with the Voight-Kampff Task @ ELOQUENT Lab in a builder-breaker style. The goal of authorship verification, as seen in PAN@CLEF2024, is to discern whether a given text is written by a human or a machine. Existing methods often perform poorly in this task due to subtle differences between human and machine-generated text, particularly as these differences become increasingly difficult to detect with the continuous improvement of generative models.

In addressing the problem of determining whether a text document is machine-generated, this paper proposes BERT_CNN: a novel approach that combines BERT and Convolutional Neural Networks (CNN) to tackle these challenges. BERT's powerful contextual understanding capabilities enable the model to capture complex patterns in the text, while CNNs excel at extracting local features such as n-gram patterns, which are crucial for distinguishing subtle differences in writing styles. By leveraging the strengths of both, our approach aims to enhance the performance of text classification tasks related to authorship verification.

Before evaluating the effectiveness of BCAV, we submitted our model to the TIRA.io[2] platform, which provides a strictly controlled testing environment to ensure fair and transparent benchmark testing against established baselines. This preliminary submission is crucial for assessing the model's applicability in real-world scenarios and improving its performance based on unbiased feedback. Following this evaluation, Bert_Transformer demonstrated outstanding performance across several key metrics. Its ROC-AUC was 0.967, Brier score was 0.79, C@1 was 0.717, F1 score was 0.719, and F0.5u was 0.717, with an overall

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

*Corresponding author.

†These authors contributed equally.

guihong163@gmail.com (G.Sun); cswyyang@fosu.edu.cn (W.Yang); molly_917@163.com (L.Ma)

0009-0007-7339-1371 (G.Sun); 0000-0003-4842-9060 (W.Yang); 0000-0002-5013-052X (L.Ma)

average score of 0.734. These results significantly outperformed other baseline models such as Unsking and Fast DetectGPT, highlighting the enhanced ability of Bert_Transformer in distinguishing between human and machine-generated text. This success underscores the effectiveness of our approach in addressing the complexity of generative AI authorship verification.

2.Background

In recent years, neural networks have seen extensive practice in text generation tasks, encompassing rule-based systems, statistical methods, and more recently, deep learning techniques. One of the most significant advancements has been the development of Transformer-based models such as GPT and BERT . An approach [3]was proposed by authors to control text generation simply, enabling researchers to adjust the generated text's theme, style, or emotional inclination without retraining the model. The CTRL [4]model emphasizes achieving finer-grained control in generation tasks by driving the model to generate text of specific styles or themes through preconditioning codes.

BERT, proposed by Devlin et al.[5], revolutionized the field of NLP by providing a robust mechanism for contextual word representation. Unlike traditional models that consider words individually, BERT captures bidirectional context, thus improving performance across a wide range of NLP tasks. However, BERT's ability to handle long-distance dependency relationships comes at the cost of sacrificing some local features, which are crucial for certain text classification tasks.

On the other hand, Convolutional Neural Networks (CNNs) are well-known for their ability to efficiently capture local patterns through convolution operations. Initially popularized in computer vision, CNNs have successfully adapted to text processing tasks, where they can effectively identify n-gram features. Some scholars[6] have reexamined CNNs' application in sentence classification tasks, particularly how fine-tuning pre-trained models and data augmentation techniques enhance performance. Others[7] have studied CNN performance across multiple text classification datasets, comparing the effects of different convolutional layer configurations and hyperparameter settings.

Combining the strengths of BERT and CNNs, recent studies[8][9][10] have explored hybrid models that leverage both global context and local features. These hybrid approaches have shown promising results but still require further exploration and validation. This study aims to fill this gap by proposing and evaluating a new BCAV hybrid model to address text generation tasks.

3. System Overview

3.1Dataset and Preprocessing

This study utilized the "Generative AI Authorship Verification Task" dataset provided by PAN@CLEF, which is a publicly available dataset specifically designed for verifying the authorship of generated AI text. The PAN@CLEF dataset contains a large number of text samples generated by different algorithms as well as text samples written by humans. These texts cover various types and topics, including news articles, blog posts, comments, and more. The PAN@CLEF dataset is typically organized into a series of text pairs, with each pair accompanied by a label indicating whether they were written by the same author. The structure specifically includes text pairs and labels, and the file formats are as follows:

{"id": "alpaca-7b/news-2021-01-01-2021-12-31-bideninauguration/art-022", "text": "..."}
{"id": "alpaca-7b/news-2021-01-01-2021-12-31-bideninauguration/art-025", "text": "..."}

The validation set provided by the "Generative AI Authorship Verification Task" in PAN@CLEF is a crucial component for testing and optimizing authorship verification models. Each text pair in the dataset is labeled as "0" or "1" to identify whether the texts are from the same author. The validation set is saved in JSONL format, with each line containing a text pair：

{"id": "VHSN9BmKWQqLABGg", "text1": "...", "text2": "..."}
{"id": "ywVXxXG12zUebGyiL8Q", "text1": "...", "text2": "..."}

In this study, the PAN@CLEF dataset was utilized to train and evaluate a hybrid model that combines BERT and CNN, with the aim of improving the accuracy of text authorship verification. Specifically, this dataset helped us systematically understand and identify the characteristics of text generated by different types of generators. Through a series of experiments on the PAN@CLEF dataset, we assessed the model's performance in distinguishing between machine-generated text and human text. We employed various evaluation metrics such as accuracy, recall, and F1 score to comprehensively analyze the model's effectiveness. The results indicate that our model can achieve satisfactory performance in handling this specific task.

3.2Network Architecture

In our research, we designed and implemented a hybrid neural network model that combines BERT and CNN to perform complex text classification tasks, specifically aimed at distinguishing between human-written and machine-generated text. The structure of this model is illustrated in Figure 1. This model architecture is intended to fully leverage BERT's deep semantic processing capabilities and CNN's local feature extraction abilities to enhance the model's performance in handling fine-grained text analysis tasks. We use the BERT-based uncased model from Hugging Face's Transformers library as our base pre-trained BERT layer. The model first processes the input text using the pre-trained BERT layer, extracting word embeddings rich in contextual information. BERT, serving as the foundational feature extractor, captures long-range dependencies in the text through its Transformer architecture. Subsequently, an attention mechanism layer is applied to the BERT output to enhance the focus and processing of critical parts of the text, thereby optimizing information flow to the subsequent layers. Multiple CNN layers then process these attention-weighted embeddings, utilizing convolutional kernels to extract local features of the text, such as n-gram patterns, which are particularly important for capturing the local semantics and style of the text. To improve the model's generalization ability on unseen data and prevent overfitting, a Dropout layer is introduced. For optimizing the model, we use a binary cross-entropy loss function to fine-tune the model's accuracy. Finally, a fully connected layer maps the output features from the CNN to classification results, determining whether a text is human-authored.

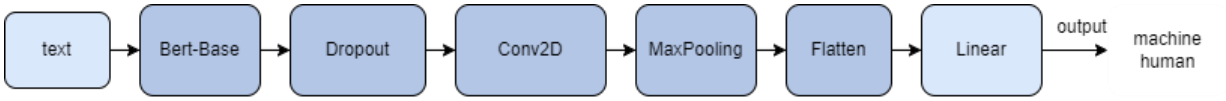

Figure 1: BCAV Architecture

4. Experiments and Results

4.1Experimental Setting

In this study, we utilized the pre-trained bert-base-uncased version of the BERT model as the foundation for text feature extraction. Due to its balanced performance and computational efficiency, it is suitable for complex text processing tasks under limited resources. We configured three sizes of convolutional kernels (3, 4, 5), each with 100 filters, to capture different lengths of n-gram features through multi-scale convolutional layers, thereby improving the model's sensitivity to local text patterns.During the training process, we chose a batch size of 8, a learning rate of 2e-5, and a total of 50 training epochs to ensure that the model could learn sufficiently and avoid overfitting. Additionally, we used the Adam optimizer, selected for its optimization effectiveness in training deep learning models, particularly in handling gradient sparsity and weight decay.To ensure the reproducibility of experiments, we

set a fixed random seed, and all experiments were conducted in a computing environment equipped with NVIDIA GeForce GTX 1660 Ti and Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz.

The main experimental process consists of three stages: data preparation, model training, and performance evaluation.Firstly, in the data preparation stage, the dataset undergoes preprocessing, including text cleaning, tokenization, and encoding using the BERT tokenizer. Additionally, the data is randomly split into training and validation sets.In the model training stage, the model iteratively learns on the training set. At the end of each epoch, the model's performance is evaluated on the validation set to monitor for overfitting during the training process.Finally, in the performance evaluation stage, we use standard classification metrics such as accuracy, ROC-AUC, etc., to evaluate the model. Special attention is given to the model's performance on an independent test set to validate its generalization ability in real-world applications.Through this series of detailed and rigorous experimental procedures, we ensure the accuracy and practicality of the research results.

4.2Result

To comprehensively evaluate the performance of our proposed BCAV, we selected a series of metrics, including ROC-AUC, Brier score, C@1, F1, and F0.5u. These metrics not only reflect the overall performance of the model but also provide different perspectives on performance evaluation, helping us to understand the model's performance in specific aspects. Additionally, we calculate the arithmetic mean of these metrics to provide a single measure for comprehensive comparison of different models.

The specific performance metrics are as follows:

ROC-AUC measures the model's ability to distinguish between positive and negative samples in a classification task. It is calculated based on integration:

$$\text{ROC-AUC} = \int_0^1 \text{TPR}(t)\ d\big(\text{FPR}(t)\big) \tag{1}$$

Brier score is used to measure the accuracy of probability predictions. It is calculated as:

$$\text{Brier Score} = \frac{1}{N}\sum_{i=1}^{N}(\text{f}_\text{i} - \text{o}_\text{i})^2 \tag{2}$$

where N is the number of samples, $f_i$is the predicted probability for sample i, and o_i is the actual label of sample i (0 or 1).

C@1 is a metric used for evaluating classification tasks. It is calculated as:

$$\text{C@1}=\frac{1}{\text{N}}\Big(\sum_{\text{i=1}}^{\text{N}}\text{I}(\widehat{y_\text{i}} = \text{y}_\text{i}) + \frac{1}{\text{N}}\sum_{\text{i=1}}^{\text{N}}\text{I}(\widehat{y_\text{i}} = -1)\sum_{\text{j=1}}^{\text{N}}\text{I}\big(\widehat{y_\text{j}} = \text{y}_\text{j}\big)\Big) \tag{3}$$

where I is the indicator function that takes a value of 1 if the condition inside the parentheses is true, otherwise 0. $\widehat{y_i}$ is the predicted label for sample i, and y_i is the actual label of sample i.

F1 score is the harmonic mean of precision and recall. It is calculated as:

$$\text{F1} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision+Recall}} \tag{4}$$

Where $\text{Precision}=\frac{\text{TP}}{\text{TP+FP}}$ $and\ Recall = \frac{\text{TP}}{\text{TP+FN}}$.

F0.5u score is a weighted F score that emphasizes precision. It is calculated as:

$$\text{F}_{0.5\text{u}} = (1 + 0.5^2) \times \frac{\text{Precision}\times\text{Recall}}{(0.5^2\times\text{Precision})+\text{Recall}} \tag{5}$$

Through these evaluation metrics, we not only assess the overall performance of the model but also conduct in-depth analysis of the model's specific performance from multiple dimensions. Ultimately, by computing the arithmetic mean of these metrics, we obtain a comprehensive performance score, which facilitates direct comparison between different

models to determine which model structure or parameter settings are more suitable for our application needs. This multi-metric evaluation approach ensures the comprehensiveness and reliability of the evaluation results, providing a solid foundation for the optimization and application of future models. Our evaluation results are shown in Table 1:

Table1：Evaluation Results Of Bert-CNN

Approach	ROC-AUC	Brier	C@1	F1	F0.5	Mean
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
BCAV	0.725	0.79	0.717	0.719	0.717	0.734

5. Conclusion

In this study, we proposed and implemented a hybrid neural network model combining BERT and CNN aimed at improving the performance of text classification tasks, particularly for authorship verification. The results indicate that the BCAV hybrid model proposed in this study achieves satisfactory results in text classification tasks. Its excellent performance across multiple evaluation metrics demonstrates the effectiveness of this model structure, providing valuable insights for future natural language processing tasks.

In future work, we will continue to refine our approach and strive for better results in authorship verification.

Acknowledgement

This work was supported by grants from the Guangdong-Foshan Joint Fund Project (No. 2022A1515140096) and Open Fund for Key Laboratory of Food Intelligent Manufacturing in Guangdong Province (No. GPKLIFM-KF-202305).

References

[1]J. Bevendorff, X. B. Casals, B. Chulvi, D. Dementieva, A. Elnagar, D. Freitag, M. Fröbe, D. Korenčić, M. Mayerl, A. Mukherjee, A. Panchenko, M. Potthast, F. Rangel, P. Rosso, A. Smirnova, E. Stamatatos, B. Stein, M. Taulé, D. Ustalov, M. Wiegmann, E. Zangerle, Overview of PAN 2024: Multi-Author Writing Style Analysis, Multilingual Text Detoxification, Oppositional Thinking Analysis, and Generative AI Authorship Verification, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2024.

[2]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.
URL:https://link.springer.com/chapter/10.1007/978-3-031-28241-6_20.
doi:10.1007/978-3-031-28241-6_20.

[3]Dathathri S, Madotto A, Lan J, et al. Plug and Play Language Models: A Simple Approach to Controlled Text Generation[J]. 2020.

[4]Keskar N S, McCann B, Varshney L R, et al. Ctrl: A conditional transformer language model for controllable generation[J]. arXiv preprint arXiv:1909.05858, 2019.

[5]Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[6]Carpuat M, de Marneffe M C, Meza-Ruiz I. Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies[C]//Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2022.
[7]Lv C, Xu J, Zheng X. Spiking convolutional neural networks for text classification[C]//The Eleventh International Conference on Learning Representations. 2022.
[8]Quoc Tran K, Trong Nguyen A, Hoang P G, et al. Vietnamese hate and offensive detection using PhoBERT-CNN and social media streaming data[J]. Neural Computing and Applications, 2023, 35(1): 573-594.
[9]Xiong G, Yan K, Zhou X. A distributed learning based sentiment analysis methods with Web applications[J]. World Wide Web, 2022, 25(5): 1905-1922.
[10]Sadat M, Caragea C. Scinli: A corpus for natural language inference on scientific text[J]. arXiv preprint arXiv:2203.06728, 2022.