Team Gladiators at PAN: Improving Author Identification: A Comparative Analysis of Pre-Trained Transformers for Multi-Author Classification

Notebook for the PAN Lab at CLEF 2024

Areeb Adnan Khan[1,*], Mohit Rai[1], Khuzaima Ali Khan[1], Syed Jahania Shah[1], Faisal Alvi[1] and Abdul Samad[1]

[1]*Dhanani School of Science and Engineering, Habib University, Karachi, Pakistan.*

Abstract

This paper presents our participation in the Multi-Author Writing Style Analysis Task for PAN at CLEF 2024. The primary goal of this task involves detecting style changes in multi-author documents at the paragraph level. The task consists of three sub-tasks: Easy, Medium, and Hard, each varying in difficulty, mainly depending on the range of topics covered in the paragraphs. We discuss the significance of style change detection in various applications such as plagiarism detection, authorship verification, and writing support. Our Approach leverages LLMs such as Electra, Deberta, Squeezebert, and Roberta.Hence we test different models to come up with the one that is most suitable for our use case, based on the F1-scores.

Keywords

Multi-Author Detector, Plagiarism Checker, Authorship Verification, Large Language Models,

1. Introduction

The PAN challenge [1] aims to identify the text positions where the author has changed based upon the author's writing style, in a multi-authored document. This challenge, along with others in [2] aims to solve it or come up with an efficient approach that would lead us to the formulation and identifying new means of detecting plagiarism, especially in cases where no-comparison text is given, as the case is with traditional plagiarism detectors. Moreover, it could be further utilized in looking for gift authorship, validating claims of authorship, and developing up-to-date writing support technologies. The dataset itself is divided into three levels of difficulty (Easy, Medium, and Hard) and each dataset contains the training, validation, and testing data [1]. The easier tasks focus on the topic information to detect the style changes, whereas the medium and hard tasks vary less in terms of topic diversity, however, they focus more on the writing style of the author to solve the task. The hard task contains the same topics. Finally, the F1-score metric is used to compare the submitted results.

2. Related Work

PAN stands for the acronym Uncovering Plagiarism, authorship, and Social Software Misuse. The Multi-Author Analysis task evolved from Author Clustering/Diarization and Author Masking/Evaluation, detailed in [3], which aimed to cluster documents by author and detect text modification. In PAN 2020 the approach commenced with two distinct datasets: narrow and wide, each accompanied by truth files delineating labels for two tasks. Initially, documents underwent paragraph segmentation to facilitate focused analysis.

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

[*] Corresponding author.

ak06865@st.habib.edu.pk (A. A. Khan); mr06638@st.habib.edu.pk (M. Rai); ka06466@st.habib.edu.pk (K. A. Khan); ss06864@st.habib.edu.pk (S. J. Shah); faisal.alvi@sse.habib.edu.pk (F. Alvi); abdul.samad@sse.habib.edu.pk (A. Samad)

0009-0007-2854-042X (A. A. Khan); 0009-0003-8881-8279 (M. Rai); 0009-0004-3334-5715 (K. A. Khan); 0009-0000-3091-694X (S. J. Shah); 0000-0003-3827-7710 (F. Alvi); 0009-0009-5166-6412 (A. Samad)

Subsequently, sentences were split, employing a nuanced approach to punctuation for accuracy. Utilizing BERT tokenization, embeddings were generated at the sentence level, with model selection guided by task specifics and performance metrics. Embedding was then amalgamated using tailored methods to suit the requirements of each task. At the document level, sentence vectors were averaged to encapsulate the document's essence. Conversely, embedding was averaged between consecutive paragraphs at the paragraph level to discern stylistic changes. Hence, this resulted in the approach of Iyer and Vosoughi [4], as performing the best in all tasks of PAN 2020.

Moreover, two different top-performing approaches for PAN 2021, Zhang et al. [5] for Task 2 and Task 3 and Strøm et al. [6] for Task 1. The former, employed ELECTRA$_{\text{Base}}$ and ELECTRA$_{\text{Large}}$ to solve all three tasks, with employing ELECTRA$_{\text{Large}}$ of max-len 128 and batch-size 64 to achieve a validation accuracy of 0.78410 for Task 2 and 0.7073 for Task 3. They pre-processed the data into paragraphs for all the models. They experimented with several batch sizes and max-lens on ELECTRA$_{\text{Large}}$ before finalizing the parameters for Task 1 and Task 3. For Task 1, Strøm et al's approach involves classifying documents as single- or multi-authored using feature extraction methods like BERT embedding and textual features. These features are processed at both the document and paragraph levels. A stacking ensemble classifier combines classifiers trained on different feature vectors. The document-level features are used for binary classification, achieving a macro F1-score of 0.7828 on the validation set and 0.7954 on the test set.

Furthermore, in PAN 2022, the top-performing [7] approach involves a unified architecture of ensemble neural networks. Lin et al. employ the approach as follows; For Task 1, which aims to identify a single style change at the paragraph level, BERT, RoBERTa, and ALBERT transformers are individually fine-tuned on labeled data, with downstream classification adjusted to binary classification for detecting style changes. Task 2, focused on assigning paragraphs to specific authors in multi-author texts, involves a similar process, with each paragraph compared to the preceding ones to determine authorship. Task 3, targeting writing style changes at the sentence level, employs the same transformer models, but fine-tuning is done using sentence pairs instead of paragraphs. The ensemble mechanism combines individual model predictions using a majority voting approach, enhancing overall detection performance across all three tasks.

Finally, in PAN 2023 the approach by Hashmi et al. [8] begins by pre-processing the data, pairing consecutive paragraphs to transform it into a multi-author evaluation task. In addition to that they combined the datasets from 2020 onwards to improve their model scores. They then use task-specific datasets to fine-tune transformer models such as BERT, RoBERTa, and ELECTRA; RoBERTa consistently performs best. By merging predictions from various models, ensemble modeling improves performance even more. In the competition evaluation, this method produces the highest F1 scores in two subtasks and second place in the third subtask, demonstrating the potency of transformer models and data augmentation in style change detection. The hyper-parameters. learning rate to 0.00001, the batch size to 16, and the number of epochs to 10.

In addition to that [9] integrates supervised contrastive learning, Rdrop, and P-tuning to improve Multi-Author Writing Style Analysis performance. It first approaches the task as binary classification, encoding using the DeBERTa-v3 model. Then, by integrating label information, supervised contrastive learning is used to enhance feature representation. Furthermore, the loss is computed for both positive and negative sample pairs using the Rdrop method. The soft-hard template is built using P-tuning, which improves the model's word embedding representation. Because it leverages pre-trained models improved by P-tuning and can capture fine-grained textual changes via supervised contrastive learning, this method works incredibly well on the hard dataset.

3. Our Approach

The dataset is neatly divided into training, validation, and test sets of data, The training set contains 70 percent of the dataset including the text files, ground-truths, and JSON files, which are used to develop our models. The validation dataset includes 15 percent of the whole dataset and the testing dataset comprises the remaining 15 percent. As the dataset is already split into relevant sections, we can carry on and consider the pairs of successive paragraphs as input samples to our model, hence concatenating them in the process, each one is then assigned a label that shows whether the data has changed or not.

Furthermore, this means that for *n* set of paragraphs in the text file, we have *n-1* paragraph pairs and Labels, hence we have converted the task of identifying the authors into binary classification, where 0 means there is no change and 1 means there is a change of authors between the paragraphs. Two such sets of datasets would be created in our case, one for training the model and the other one for validation of the model.

Table 1
Total Number of changes will be used for Binary Classification

Data Types	Total Number of changes in authors ([1]) for consecutive paragraphs	Total number of no changes in authors ([0]) for consecutive paragraphs
Easy	10094	966
Medium	12491	9419
Hard	8915	10094

In Table 1, it is evident that the easy dataset exhibits class imbalance, with a significantly higher occurrence of class 1 (left column) compared to class 0 (right column). To address this imbalance, we implemented a weighted random sampler. To implement this we first ensure that the **'changes'** column in the training data frame is appropriately formatted. It then computes the class distribution by counting the occurrences of each class label. By inversely weighting the class frequencies, the function calculates a set of weights such that minority class samples are given a higher probability during sampling. These weights are then used to create a weighted random sampler, which ensures that each class is represented proportionally in the training process, thereby mitigating the effects of class imbalance and potentially improving model performance on minority classes.

3.1. Pre-Trained Transformer Models

Pre-trained transformer models have been trained on enormous volumes of text data. They gain an understanding of language's structure and patterns, which helps them produce text of the highest caliber and carry out various natural language processing (NLP) functions. By fine-tuning the pre-trained models, we can take advantage of their language comprehension abilities and apply the knowledge they have learned from their thorough pre-training to our particular task. Hence for this project, we employed popular pre-transformer models RoBERTa, ELECTRA, DeBERTa, and SqueezeBERT.

RoBERTa uses dynamic masking and omits next-sentence prediction to improve BERT's pre-training, resulting in stronger word representations. **ELECTRA** focuses on efficiency, using synthetic data with replaced words to teach the model to distinguish them from originals, achieving good results with less training data. **DeBERTa** combines ideas from ELECTRA and BERT, separating word and positional information for better context understanding and using an ELECTRA-like task for effective training and high performance. **SqueezeBert** reduces model size by compressing a large pre-trained model like BERT into a faster, smaller version while preserving most of its functionality through knowledge distillation

3.2. Preliminary Results

We used the base versions of several pre-trained models available on HuggingFace. These models were implemented on Kaggle notebooks, and they were then refined further. We used the following hyperparameter values to improve the model's performance: a maximum sequence length of 256, a learning rate of 0.00001, a batch size of 16, and 12 epochs.

We calculated the F1 score on the given evaluation set to assess the models' effectiveness for every subtask. This F1 score was determined by comparing the models' predictions for each sub-task evaluation set's ability to identify style changes between consecutive paragraph pairs.

Table 2
F1-scores for different models on different datasets

Electra-base-discriminator F1-score		
Data Types	**Training Data-Set**	**Validation Data-Set**
Easy Dataset	0.947	0.939
Medium Dataset	0.974	0.802
Hard Dataset	0.973	0.713

Deberta-v3-base F1-score		
Data Types	**Training Data-Set**	**Validation Data-Set**
Easy Dataset	0.932	0.921
Medium Dataset	0.899	0.789
Hard Dataset	0.871	0.756

squeezebert-mnli-headless F1-score		
Data Types	**Training Data-Set**	**Validation Data-Set**
Easy Dataset	0.852	0.773
Medium Dataset	0.810	0.700
Hard Dataset	0.791	0.680

Roberta-base F1-score		
Data Types	**Training Data-Set**	**Validation Data-Set**
Easy Dataset	0.988	0.940
Medium Dataset	0.991	0.806
Hard Dataset	0.981	0.761

3.3. Analysis

From the results shown in Table 2, we observe a distinct trend in model performance. On the easy dataset, the RoBERTa model achieves the highest F1-score of 0.940, followed closely by ELECTRA with an F1-score of 0.939. DeBERTa secures the third position with an F1-score of 0.756, while SqueezeBERT ranks last. A similar performance pattern is evident in the medium dataset, with RoBERTa leading and SqueezeBERT trailing.

However, in the hard dataset, the performance dynamics shift. Although RoBERTa maintains its leading position, DeBERTa surpasses ELECTRA, achieving an F1-score of 0.756 compared to ELECTRA's 0.713. SqueezeBERT continues to underperform in this dataset as well.

The observed trend in model performance across the Easy, Medium, and Hard datasets can be attributed to several factors related to the models' inherent architecture and training strategies.

Thanks to a dynamic masking method, intensive training on large-scale datasets, and a resilient architecture, RoBERTa regularly outperforms competing models. These elements enable RoBERTa to more successfully identify complex patterns in the data, as seen by its high F1 scores in every dataset. ELECTRA, on the other hand, is comparable to RoBERTa. Still, its distinct pre-training method—which uses substitute token detection—might not be able to handle the subtle complexity seen in more difficult datasets. This explains the modest decline in ELECTRA's performance compared to RoBERTa, especially in the Hard dataset.

Different datasets show different performances for DeBERTa, which uses relative position embeddings and disentangled attention mechanisms. While it does well overall, its architecture may not fully take advantage of the simpler structures in the Easy and Medium datasets, which could explain why it ranks third in these situations while SqueezeBERT's lower performance across all datasets can be attributed to its design, which prioritizes model efficiency and reduced computational resources over capturing complex patterns.

Based upon the results it was best to work with **RoBERTa** and fine-tune the model further to achieve better scores.

3.4. Revised Strategy

To further improve the performance metrics of the **RoBERTa** model, we initially considered utilizing the larger variant available on HuggingFace. However, given the widespread use of large language models (LLMs) in the current research community and their significant environmental impact, we opted for alternative optimization methods. These approaches are environmentally sustainable and capable of achieving comparable performance enhancements.

One way we thought of improving the performance was by increasing the size of the dataset, this method is also called data augmentation. To realize this we had a unique approach since we were giving models two paragraphs and a label. Let's consider paragraphs A and B with a label of 0, indicating no change. We pass this data to our models, but if we reverse the order of the paragraphs—placing paragraph B before paragraph A—the label remains unchanged. This method effectively doubles the size of our dataset.

Table 3
Change in dataset size

Data Types	Previous Dataset Size	New Dataset Size
Easy	11062	22124
Medium	21911	43822
Hard	19010	38020

In Table 3 we can see the increase in the size of the dataset and we re-train **RoBERTa** base model with the same experimental settings as previously mentioned and obtained the following results

Table 4
F1-scores for **RoBERTa** on Revised Dataset

RoBERTa-base-discriminator F1-score		
Data Types	**Training Dataset**	**Validation Dataset**
Easy Dataset	0.989	0.958
Medium Dataset	0.987	0.816
Hard Dataset	0.983	0.787

As demonstrated in Table 4, employing the new strategy has improved scores across all datasets. This suggests that augmenting the dataset size positively impacts performance metrics. Specifically, the Easy dataset achieved an F1-score of 0.958, the Medium dataset reached an F1-score of 0.816, and the Hard dataset attained an F1-score of 0.787. Based on these results, we conclude that the **RoBERTa** base models, trained on the Easy, Medium, and Hard datasets, should be submitted for evaluation on the TIRA platform.

4. Results

The approaches titled "presto-door" and "null-directory" are the same models, i.e. both are the score obtained through the same Roberta-base-discriminator model. The only difference between the approaches is that one was submitted manually through Docker, whereas the other was submitted through the GitHub Actions tool. Moreover, test set results on approaches titled "presto-door" and "null-directory" displayed in Table 5 reiterate our revised strategy of expanding the dataset and then utilizing the Roberta model on it. The scores for the test sets are much better and closer to the validation set provided. In conclusion, this validates that, first of all, Roberta is the best-performing model amongst the models

Table 5
Overview of the F1 scores for the multi-author writing style task in detecting at which positions the author changes for task 1, task 2, and task 3 on the **test set**

Approach	Task 1	Task 2	Task 3
presto-door	0.956	0.809	0.783
null-directory	0.956	0.809	0.783
Baseline Predict 1	0.466	0.343	0.320
Baseline Predict 0	0.112	0.323	0.346

tested, and secondly the strategy of Data Augmentation aids in providing a better model F1-scores than, the models trained on just the initial datasets provided.

As students, academians, and part of a community that is aimed towards protecting the environment, building upon the United Nations sustainability goals, and gearing towards lower greenhouse emissions, we choose to work with models that have minimal impact on the environment, as we efficiently tried to utilize the resources available to us, to complete the task of multi-author classification in PAN 2024

5. Conclusion

Our solution builds upon binary classification, hence it works by treating the Paragraph as two instances of input text and giving output a label that tells whether the author is changing or not. While our approach achieved high accuracy on the task of multi-author classification using the best models out of the many fine-tuned transformers, it's important to acknowledge some limitations. Running multiple models can be computationally expensive, and fine-tuning often requires significant labeled data. Additionally, the complex nature of transformer models can make it difficult to understand how they arrive at their predictions. Despite these limitations, our approach achieved strong performance by leveraging the best-performing models based on F1 scores on both the training and test sets. However, it's important to note that this binary classification approach treats the task as a simple presence or absence of a style change, neglecting the potential for nuanced stylistic variations within a single author's work. Future work could explore multi-class classification to capture a wider range of stylistic distinctions or delve deeper into interpretability techniques to understand the reasoning behind the models' predictions.

Acknowledgments

The authors would like to acknowledge the support provided by the Office Of Research (OoR) at Habib University, Karachi, Pakistan for funding this project through the internal research grant IRG-2235.

References

[1] E. Zangerle, M. Mayerl, M. Potthast, B. Stein, Overview of the Multi-Author Writing Style Analysis Task at PAN 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-WS.org, 2024.

[2] 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: 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.

[3] P. Rosso, F. Rangel, M. Potthast, E. Stamatatos, M. Tschuggnall, B. Stein, Overview of PAN 2016–New Challenges for Authorship Analysis: Cross-genre Profiling, Clustering, Diarization, and Obfuscation, in: N. Fuhr, P. Quaresma, B. Larsen, T. Gonçalves, K. Balog, C. Macdonald, L. Cappellato, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. 7th International Conference of the CLEF Initiative (CLEF 2016), volume 9822 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2016, pp. 518–538. doi:10.1007/978-3-319-44564-9_28.

[4] A. Iyer, S. Vosoughi, Style change detection using BERT: Notebook for PAN at CLEF 2020 (2020).

[5] X. Jiang, H. Qi, Z. Zhang, M. Huang, Style change detection: Method based on pre-trained model and similarity recognition, in: Notebook Papers of PAN at CLEF 2022, Foshan, China, 2022.

[6] E. Strøm, Multi-label style change detection by solving a binary classification problem, in: Notebook Papers of PAN at CLEF 2021, Høgskoleringen 1, 7491 Trondheim, Norway, 2021.

[7] T.-M. Lin, C.-Y. Chen, Y.-W. Tzeng, L.-H. Lee, Ensemble pre-trained transformer models for writing style change detection, in: Notebook Papers of PAN at CLEF 2022, National Central University, Taiwan, 2022.

[8] A. Hashemi, W. Shi, Enhancing writing style change detection using transformer-based models and data augmentation, in: Notebook Papers of PAN at CLEF 2023, Ottawa, Canada, 2023.

[9] Z. Ye, C. Zhong, H. Qi, Y. Han, Supervised contrastive learning for multi-author writing style analysis, Department of Electrical Engineering, Foshan University, China (2023).