Team TMU at PAN 2025: An Ensemble of Fine-Tuned LaBSE and Siamese Neural Network for Multi-Author Writing Style Analysis

Notebook for PAN at CLEF 2025

Sara Bourbour Hosseinbeigi[1,*], Ali Mehrani[1]

[1]Tarbiat Modares University, Tehran, Iran

Abstract

This study suggests an ensemble-based approach to tackle the PAN 2025 Multi-Author Writing Style Analysis task, which necessitates the identification of stylistic variations within a text by examining sentence pairs to ascertain authorship similarity. We suggest two models that have been trained on sentence pairs to tackle this challenge. First, we fine-tune the LaBSE model using labeled pairs, where each label denotes a possible authorship change. We create a feature vector for each pair that contains the original LaBSE embeddings of both sentences, their absolute differences, and directional cross-attention outputs showing the relationship between the two sentences. In our second approach, we train a Siamese neural network consisting of two Bi-LSTMs on the same sentence pairs, using their token-level embeddings generated by FastText as input to predict authorship change. Finally, we use an XGBoost classifier to put the two models together in order to further enhance our performance. In the test sets for the easy, medium, and high difficulty levels, we obtained F1 scores of 0.95, 0.792, and 0.792, respectively.

Keywords

PAN 2025, Multi-Author Writing Style Analysis, Pre-trained Models, LaBSE Model, Siamese Neural Network, Ensemble Learning

1. Introduction

Multi-author writing style analysis refers to the task of identifying stylistic differences within a document written by multiple authors. Its goal is to detect the points in which a shift in authorship is indicated by a change in writing style. Applications such as authorship attribution and plagiarism detection benefit from this study. It typically involves modeling sentence- or paragraph-level features to determine whether different parts of a text were written by the same or different individuals [1].

Since 2016, PAN has hosted an annual challenge focused on analyzing multi-author documents, aiming to detect where the writing style changes within a text [2]. The 2025 edition of the Style Change Detection (SCD) task by PAN focuses on identifying writing style changes at the sentence level. Given a multi-author document, the objective is to locate the points where authorship changes by analyzing shifts in writing style [1, 3]. The task provides datasets at three difficulty levels: easy, medium, and hard, each requiring participants to identify style change positions. The easy dataset features diverse topics, while the medium and hard datasets contain limited or no topic variation [1, 4].

In this paper, we propose a solution to the 2025 SCD task that combines semantic and morphological representations. Our method fine-tunes the LaBSE (Language-agnostic BERT Sentence Embedding) [5] model to capture semantic relationships and uses a Siamese BiLSTM network [6, 7] to extract morphological and surface-level structural patterns. An XGBoost classifier [8] is then used to ensemble their outputs and generate the final predictions.

Contribution. Our main contribution is a hybrid framework that effectively captures both semantic and morphological features to detect authorship changes at the sentence level. We fine-tune LaBSE to

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

*Corresponding author.

s.bourbour@modares.ac.ir (S. B. Hosseinbeigi); ali.mehrani@modares.ac.ir (A. Mehrani)

extract rich semantic features and design a Siamese BiLSTM model trained on FastText [9] embeddings to focus on writing style differences. By ensembling these two perspectives using an XGBoost classifier, our approach achieves more robust and accurate predictions.

2. Related Work

The 2022 edition of the Style Change Detection (SCD) task featured three sub-tasks, focusing on detecting authorship changes at both the paragraph and sentence levels, while also assigning each paragraph to a specific author from among the assumed authors [10]. The top-performing approach by Lin et al. [11] fine-tuned three transformer models—BERT, RoBERTa, and ALBERT—and combined their outputs using majority voting to generate the final predictions. Other high-performing approaches in the 2022 SCD task, also relied on fine-tuning pre-trained language models such as BERT in [12] and ELECTRA in [13].

The 2023 edition of the SCD task focused on identifying writing style changes at the paragraph level, using datasets of three difficulty levels: easy, medium, and hard. The easy dataset featured diverse topics, whereas the medium and hard datasets contained limited or no topic variation [14]. Hashemi and Shi [15] fine-tuned BERT, RoBERTa, and ELECTRA separately, and combined their predictions using majority voting. They also employed data augmentation techniques to achieve the best results on the easy and medium sets of the 2023 SCD task. Other high-performing approaches, such as [16], [17], and [18], also relied on using pre-trained language models like DeBERTa and mT0-xl.

In the 2024 SCD task, participants were asked to identify all paragraph-level positions where the writing style changes within a given text. As in the 2023 SCD task, the datasets were divided into three difficulty levels: easy, medium, and hard, with the latter two featuring little to no topic diversity [2]. Most approaches in the 2024 SCD task relied on pre-trained language models, with a strong preference for models from the BERT family. For instance, several teams utilized RoBERTa, DeBERTa, or both [19, 20, 21], while another team experimented with LLaMA 3 [22], showing a continued trend toward using powerful transformer-based models for style change detection.

Based on previous editions of the SCD task, it's clear that pre-trained language models have played a key role in writing style analysis. Their ability to capture context and provide strong sentence-level representations makes them an effective choice for downstream tasks such as style change detection. In our work, we chose to use LaBSE as the backbone encoder. Although LaBSE was originally developed for multilingual applications, it also performs very well on English data, offering semantically meaningful embeddings suitable for comparing text segments [5]. To better align it with our task, we fine-tuned it on our training data so it could capture the semantic differences between sentences. We further describe this fine-tuned LaBSE model, along with our Siamese network, in detail in the *Methodology* section.

3. Methodology

Our approach to sentence-level authorship change detection is based on combining semantic and morphological information through two complementary models. First, we fine-tune the LaBSE model to capture semantic relationships between consecutive sentences. Second, we train a Siamese BiLSTM model on FastText embeddings to focus on surface-level and morphological differences. The predictions from both models are then combined using an XGBoost classifier, which serves as the final decision layer. This ensemble makes use of both deep semantic encoding and surface-level morphological modeling, improving the system's performance, especially on harder cases where there is little to no topic variation. In the following subsections, we describe our data processing pipeline, the LaBSE fine-tuning strategy, the Siamese network architecture, and the ensembling method used to combine model predictions. Implementation details are available at https://github.com/alimrn001/PAN-2025-Authorship-Change-Detection.

3.1. Data Processing

In this year's SCD task, the dataset is divided into three difficulty levels: easy, medium, and hard. Each subset consists of documents composed of multiple sentences [4]. To prepare the data, we process each document by generating sentence pairs from consecutive sentences, resulting in *n−1* pairs for a document containing *n* sentences. Table 1 presents statistics for the training and validation sets across the dataset's easy, medium, and hard subsets, including the total number of documents, the number of generated sentence pairs, and the label distribution within each set.

Table 1
Statistics of the training and validation sets across the dataset's easy, medium, and hard subsets, including the total number of documents, sentence pairs, and label distribution.

Dataset	Total Docs	Total Pairs	Positive Samples	Negative Samples	Approx. Distribution
Easy - Train	4200	48402	10224	38178	0.21 / 0.79
Easy - Validation	900	10247	2201	8046	0.21 / 0.79
Medium - Train	4200	58817	12503	46314	0.21 / 0.79
Medium - Validation	900	12759	2750	10009	0.21 / 0.79
Hard - Train	4200	51061	8815	42246	0.17 / 0.83
Hard - Validation	900	10648	1906	8742	0.18 / 0.82

3.2. Fine-tuning the LaBSE Pre-trained Language Model

To capture semantic shifts that may indicate authorship changes, we fine-tune the LaBSE model using a sentence-pair classification setup. Each input is a pair consisting of two consecutive sentences from a document, labeled as either indicating a style change or not. Instead of relying only on LaBSE's sentence embeddings, we apply a feature fusion mechanism for our classification task. At first, we concatenate the absolute difference of the two sentence embedding vectors with the embeddings themselves. Later, we employ a more expressive cross-attention fusion mechanism. Our architecture enhances LaBSE by incorporating a lightweight attention-based interaction layer that allows each sentence's representation to be informed by the other.

To capture interactions between the sentences, we apply bidirectional cross-attention. Specifically, the embedding of sentence 1 (`S1`) attends over all token embeddings of sentence 2, and vice versa. This procedure generates two new vectors called `Cross12` and `Cross21`, capturing how each sentence "sees" the other. To construct our final feature vector, we concatenate the following components and fine-tune LaBSE on these feature vectors: (1) the embeddings of both sentences, named `S1` and `S2`; (2) the absolute difference between `S1` and `S2` (`|S1 - S2|`); and (3) the two cross-attention vectors, named `Cross12` and `Cross21`. This results in a feature vector of dimension $5 \times$ `embed_dim`, where `embed_dim` represents the size of an embedding vector generated by LaBSE and is equal to 768.

After the final feature vector is constructed for each pair of sentences, it is fed into a classification head consisting of a dropout layer (with a dropout rate of 0.1) and a fully connected linear layer that directly outputs logits for the two classes (0 and 1).

For the fine-tuning process, we use a weighted cross-entropy loss to address class imbalance, where the class weights are computed based on the label distribution in the training data. Our LaBSE model is fine-tuned for 3 epochs using the AdamW optimizer with a learning rate of 2×10^{-5} and a batch size of 32. All parameters of the LaBSE model, including the backbone, remain unfrozen and are updated during fine-tuning.

3.3. The Siamese Neural Network

First introduced by Bromley et al. [6], Siamese neural networks are a type of architecture that consists of two (or more) identical sub-networks that share the same weights, each processing one of the input vectors independently. The outputs of these sub-networks are later compared using a similarity function,

such as cosine similarity. The final output of the Siamese network shows how similar or dissimilar the two inputs are, making it effective for tasks that require similarity assessment. Siamese neural networks have been applied in different fields, including audio processing, image recognition, and text mining [23].

To capture morphological and stylistic differences between sentences, we design a Siamese neural network that complements the semantic modeling of the LaBSE model. Our Siamese model uses a pair of BiLSTM sub-networks, which process FastText-based token embeddings for each sentence independently. The model is trained to detect whether two consecutive sentences in a document represent a change in writing style, indicating an authorship change.

Each sentence is first tokenized and converted into a sequence of word embeddings using the pre-trained FastText model. These embeddings have a dimension of 300 and capture both semantic and subword-level information. For each sentence, we limit the number of tokens to a maximum length of 100. A sentence is zero-padded if shorter than this length, and truncated to 100 tokens if longer.

Our model's architecture consists of two BiLSTM sub-networks, a pairwise comparison unit, and a classification head. Each sentence is encoded into a 256-dimensional vector by passing its FastText embeddings through the shared BiLSTM, which captures both forward and backward context. Given these two sentence representations, the model computes their absolute difference, which captures their surface-level and stylistic differences. This difference vector is then passed through a classification head, which is a feedforward neural network with a linear layer with ReLU activation and a dropout rate of 0.3, followed by a final linear layer that outputs logits (raw scores) for two classes, indicating whether a style change occurs between the input sentences.

Our Siamese model uses a weighted cross-entropy loss to handle class imbalance and is trained for 3 epochs, using the Adam optimizer with a learning rate of 10^{-3}.

3.4. Ensembling the Predictions

To combine the strengths of our models, we implement an XGBoost-based ensembling approach. For each sentence pair, we generate the probability vectors from both of our models, and concatenate them to create a feature vector of size 4. We then train an XGBoost classifier on these feature vectors, which is trained on the training set, while the validation set is used for evaluation. Our XGBoost classifier achieved better results compared to both of our models, especially on the medium and hard subsets where topic variation is minimal, as we discuss in detail in Section 4.

Figure 1 shows the architecture of our system for predicting writing style changes between two sentences. First, the raw sentences are given as input to the LaBSE model to generate sentence embeddings. For fine-tuning LaBSE, pairs of consecutive sentence embeddings are combined into a feature vector, which is then passed through a classification layer on top of LaBSE and fine-tuned for the style change detection task. Simultaneously, token-level embeddings of the raw sentences are generated using FastText. These embeddings are fed into a Siamese network containing two BiLSTM encoders, which are trained with each pair's corresponding label for style change. Both models are trained for 3 epochs independently. After their training is finished, the outputs (raw logits) from the LaBSE and Siamese BiLSTM models are first transformed into probability scores using the softmax function. These probability vectors are then concatenated to create a feature vector of size 4 for each sentence pair. This feature vector is then used as input to train the XGBoost classifier, which produces the final predictions for the style change detection task.

4. Evaluation

We trained and evaluated our models on the datasets provided by the task organizers, which is available in three difficulty levels of easy, medium, and hard [4]. To generate token-level embeddings for our Siamese model, we used pre-trained FastText word vectors, which were downloaded from its official

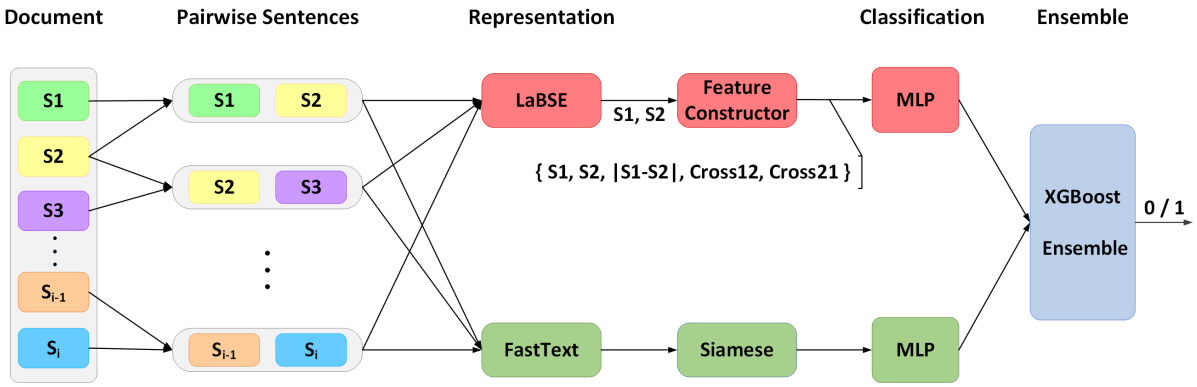

Figure 1: The architecture of our proposed system, which creates a pairwise dataset from the input document, fine-tunes LaBSE, trains a Siamese network on FastText representations, and finally ensembles the outputs using an XGBoost classifier. The *Feature Constructor* module uses *Differentiation* and *Cross-Attention* submodules to build the final feature vector, consisting of the embeddings of two consecutive sentences, their absolute difference, and two cross-attentive vectors.

website[1]. To acquire the LaBSE model, we used the Sentence Transformers library [24], which provides access to a variety of pre-trained models.

4.1. Settings

To implement our system, we used the PyTorch framework and conducted our experiments on an NVIDIA RTX A6000 GPU. Our LaBSE model is fine-tuned for 3 epochs using labeled sentence pairs, where each sentence is tokenized to a maximum length of 512 tokens. The model uses the AdamW optimizer with a learning rate of 2×10^{-5}, a batch size of 32, and a dropout rate of 0.1.

Our Siamese model was also trained for 3 epochs using labeled sentence pairs where each sentence is fed into a BiLSTM network with a hidden size of 128 in each direction, resulting in a 256-dimensional sentence representation. Inputs to the BiLSTM sub-network are 300-dimensional word embeddings generated by FastText, with each sentence tokenized and either padded or truncated to a fixed length of 100 tokens. The model was trained using the Adam optimizer with a learning rate of 10^{-3}, a batch size of 32, and a dropout rate of 0.3.

Our XGBoost classifier was trained using 100 boosting rounds, a maximum tree depth of 3, and a learning rate of 0.1, and was optimized with the log-loss evaluation metric.

4.2. Results

Table 2 shows the evaluation results of our models on the validation set, which includes easy, medium, and hard subsets. As can be seen, the fine-tuned LaBSE model performs strongly on the easy set on its own. However, on the medium and hard sets, our ensembling approach significantly outperforms both the LaBSE and Siamese models individually. This indicates that our ensemble successfully complements and combines the strengths of both models, capturing both semantic and morphological differences and similarities between sentence pairs. This is because LaBSE (as a transformer-based pre-trained language model) is highly effective at capturing high-level semantic relationships [5]. In contrast, our Siamese BiLSTM model uses FastText word embeddings and focuses on details like word choice and structural patterns, which show an author's writing style. Therefore, combining these two approaches can improve the accuracy of authorship change detection, especially when the topics are similar.

To evaluate our system on the final test set not seen by the model, we uploaded our models to HuggingFace and submitted our approach to TIRA [25]. Table 3 shows the evaluation results (F1 scores) of our approach on the test dataset on the easy, medium, and hard subsets. As can be seen, our model has achieved the F1 score of 0.95 on the easy subset and 0.792 on the medium and hard subsets.

[1] https://fasttext.cc

Table 2
F1 scores of the fine-tuned LaBSE model, the Siamese BiLSTM network, and their XGBoost ensemble, evaluated on the validation set across three difficulty levels: easy, medium, and hard.

	Siamese	LaBSE	Ensemble
Easy	0.857	0.941	0.939
Medium	0.762	0.782	0.800
Hard	0.711	0.769	0.781

Table 3
F1 scores of the XGBoost ensemble model evaluated on the unseen test set (via TIRA) across three difficulty levels: easy, medium, and hard.

	Ensemble
Easy	0.95
Medium	0.792
Hard	0.792

5. Conclusion and Future Work

In this paper, we proposed a hybrid ensemble-based approach for the PAN 2025 Style Change Detection task. Our architecture integrated the semantic capabilities of a fine-tuned LaBSE model with the morphological capabilities of a Siamese BiLSTM network trained on FastText embeddings. By combining the outputs of these two models through an XGBoost classifier, we achieved strong performance across datasets of different difficulty levels.

To further improve our approach, we can explore several techniques to improve our model's accuracy. For example, when fine-tuning LaBSE, we can enrich the feature vectors by adding extra contextual information, such as text complexity or sentiment. Additionally, we can explore LLM-based methods like LLM-as-a-Judge, using the in-context learning capabilities of large language models.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT for grammar and spelling checks. After using this tool, the authors reviewed and edited the content, and take full responsibility for the final publication.

References

[1] J. Bevendorff, D. Dementieva, M. Fröbe, B. Gipp, A. Greiner-Petter, J. Karlgren, M. Mayerl, P. Nakov, A. Panchenko, M. Potthast, A. Shelmanov, E. Stamatatos, B. Stein, Y. Wang, M. Wiegmann, E. Zangerle, Overview of PAN 2025: Generative AI detection, multilingual text detoxification, multi-author writing style analysis, and generative plagiarism detection - extended abstract, in: C. Hauff, C. Macdonald, D. Jannach, G. Kazai, F. M. Nardini, F. Pinelli, F. Silvestri, N. Tonellotto (Eds.), Advances in Information Retrieval - 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6-10, 2025, Proceedings, Part V, volume 15576 of *Lecture Notes in Computer Science*, Springer, 2025, pp. 434–441. URL: https://doi.org/10.1007/978-3-031-88720-8_64. doi:10.1007/978-3-031-88720-8_64.

[2] 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. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 2424–2431. URL: https://ceur-ws.org/Vol-3740/paper-222.pdf.

[3] E. Zangerle, M. Mayerl, M. Potthast, B. Stein, Overview of the multi-author writing style analysis task at PAN 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2025.

[4] E. Zangerle, M. Mayerl, M. Potthast, B. Stein, PAN25 multi-author writing style analysis, 2025. URL: https://doi.org/10.5281/zenodo.14891240. doi:10.5281/zenodo.14891240.

[5] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, W. Wang, Language-agnostic BERT sentence embedding, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, Association for Computational Linguistics, 2022, pp. 878–891. URL: https://doi.org/10.18653/v1/2022.acl-long.62. doi:10.18653/V1/2022.ACL-LONG.62.

[6] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a "Siamese" time delay neural network, in: J. D. Cowan, G. Tesauro, J. Alspector (Eds.), Advances in Neural Information Processing Systems 6, [7th NIPS Conference, Denver, Colorado, USA, 1993], Morgan Kaufmann, 1993, pp. 737–744. URL: http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network.

[7] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing 45 (1997) 2673–2681. URL: https://doi.org/10.1109/78.650093. doi:10.1109/78.650093.

[8] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: B. Krishnapuram, M. Shah, A. J. Smola, C. C. Aggarwal, D. Shen, R. Rastogi (Eds.), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, ACM, 2016, pp. 785–794. URL: https://doi.org/10.1145/2939672.2939785. doi:10.1145/2939672.2939785.

[9] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146. URL: https://doi.org/10.1162/tacl_a_00051. doi:10.1162/TACL_A_00051.

[10] E. Zangerle, M. Mayerl, M. Potthast, B. Stein, Overview of the style change detection task at PAN 2022, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 2344–2356. URL: https://ceur-ws.org/Vol-3180/paper-186.pdf.

[11] T. Lin, C. Chen, Y. Tzeng, L. Lee, Ensemble pre-trained transformer models for writing style change detection, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 2565–2573. URL: https://ceur-ws.org/Vol-3180/paper-210.pdf.

[12] Q. Lao, L. Ma, W. Yang, Z. Yang, D. Yuan, Z. Tan, L. Liang, Style change detection based on Bert and Conv1d, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 2554–2559. URL: https://ceur-ws.org/Vol-3180/paper-208.pdf.

[13] X. Jiang, H. Qi, Z. Zhang, M. Huang, Style change detection: Method based on pre-trained model and similarity recognition, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 2526–2531. URL: https://ceur-ws.org/Vol-3180/paper-205.pdf.

[14] E. Zangerle, M. Mayerl, M. Potthast, B. Stein, Overview of the multi-author writing style analysis task at PAN 2023, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2513–2522. URL: https://ceur-ws.org/Vol-3497/paper-201.pdf.

[15] A. Hashemi, W. Shi, Enhancing writing style change detection using transformer-based models

and data augmentation, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2613–2621. URL: https://ceur-ws.org/Vol-3497/paper-212.pdf.

[16] H. Chen, Z. Han, Z. Li, Y. Han, A writing style embedding based on contrastive learning for multi-author writing style analysis, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2562–2567. URL: https://ceur-ws.org/Vol-3497/paper-206.pdf.

[17] I. E. Kucukkaya, U. Sahin, C. Toraman, ARC-NLP at PAN 2023: Transition-focused natural language inference for writing style detection, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2659–2668. URL: https://ceur-ws.org/Vol-3497/paper-218.pdf.

[18] M. Huang, Z. Huang, L. Kong, Encoded classifier using knowledge distillation for multi-author writing style analysis, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2629–2634. URL: https://ceur-ws.org/Vol-3497/paper-214.pdf.

[19] T. Lin, Y. Wu, L. Lee, NYCU-NLP at PAN 2024: Integrating transformers with similarity adjustments for multi-author writing style analysis, in: G. Faggioli, N. Ferro, P. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 2716–2721. URL: https://ceur-ws.org/Vol-3740/paper-255.pdf.

[20] Y. Huang, L. Kong, Team text understanding and analysis at PAN: Utilizing BERT series pre-training model for multi-author writing style analysis, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. S. Herrera (Eds.), Working Notes Papers of the CLEF 2024 Evaluation Labs, CEUR-WS.org, 2024, pp. 2653–2657. URL: http://ceur-ws.org/Vol-3740/paper-246.pdf.

[21] Z. Huang, L. Kong, Deberta-v3 with r-drop regularization for multi-author writing style analysis, in: G. Faggioli, N. Ferro, P. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 2658–2664. URL: https://ceur-ws.org/Vol-3740/paper-247.pdf.

[22] J. Lv, Y. Yi, H. Qi, Team fosu-stu at PAN: supervised fine-tuning of large language models for multi author writing style analysis, in: G. Faggioli, N. Ferro, P. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 2781–2786. URL: https://ceur-ws.org/Vol-3740/paper-265.pdf.

[23] D. Chicco, Siamese neural networks: An overview, in: H. M. Cartwright (Ed.), Artificial Neural Networks - Third Edition, volume 2190 of *Methods in Molecular Biology*, Springer, 2021, pp. 73–94. URL: https://doi.org/10.1007/978-1-0716-0826-5_3. doi:10.1007/978-1-0716-0826-5_3.

[24] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese BERT-networks, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, Association for Computational Linguistics, 2019, pp. 3980–3990. URL: https://doi.org/10.18653/v1/D19-1410. doi:10.18653/V1/D19-1410.

[25] 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: 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.