Different Encoding Approaches for Authorship Verification

Stefanos Konstantinou[1], Jinqiao Li[1] and Angelos Zinonos[1]

[1]University of Zurich, Rämistrasse 71, 8006 Zürich, Switzerland

Abstract

PAN is a series of scientific events and shared tasks which focuses on digital text forensics and stylometry. In previous editions of PAN, the effectiveness of authorship verification technology in several languages and text genres was tackled, and this year the content shifted to cross discourse type pairs of text. The purpose of this paper is to test various Transformer based encoder models, using Cross encoder and Bi-Encoder approaches. The results illustrate a decent performance, reaching an F1 score of 80% on the best model. Further experimentation was performed on the training dataset, which resulted in no positive outcome.

Keywords

NLP, Author Verification, PAN22, Pre-trained model, Text information, Classification

1. Introduction

This paper presents our approach for the Authorship Verification Shared Task [1] at PAN 2022 [2]. The goal of the task is to decide whether two texts have been written by the same author based on comparing their writing styles. Compared to the tasks in previous editions, this year's aim is to focus on more challenging scenarios, as this will allow studying the ability of stylometric approaches to capture authorial characteristics even when different discourse types are imposed. Discriminating between documents by stylometric means, could indicate significant boosts in the area of Cyber Security and Criminology. Moreover, people producing or writing hate speech on social platforms anonymously could be identified from even their business emails if the task is successfully solved thus, they can be held accountable.

The dataset contains essays, emails, text messages and business memos in English. The purpose of this task is to develop a method that will compare a pair of texts consisting of different discourse types and to predict whether they are written by the same author or not.

After analyzing related work, we have decided to follow a transformer-based approach since it is widely and successfully used in Natural Language Processing. Our goal is to experiment with a wider variety of transformer models than what has been tested before.

2. Related Work

[3] gives an overview of the approaches from last year's instance of the PAN CLEF author

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy

stefanos.konstantinou@uzh.ch (S. Konstantinou); jinqiao.li@uzh.ch (J. Li); angelos.zinonos@uzh.ch (A. Zinonos)

verification task. Successful methods and their maximum performance on an identical task and evaluation metric as this year's task are presented, with differences in context in the data set. [4] concludes that best performance is achieved by combining heterogeneous methods, for instance, applying machine learning techniques such as decision trees and artificial neural networks.

[5] aims to analyze the problem of correlating the author's characteristics with the attributes of documents written by that same author. They highlight that the first step should be identifying the essential feature in the text to conduct a better analytic phase. Once the relevant features are extracted from a document, different methods can be experimented with to identify the author. This was useful for their implementation of SVMs and Random Forests.

Transformer-based architectures have been experimented with on earlier versions of this task, but were limited to using BERT. In [6], a pre-trained model of Bert was presented as a solution for encoding text information of text pairs. Data-record splitting was introduced to create short texts that can be encoded by BERT, achieving the highest c@1 and F1-score on PAN Authorship Verification datasets.

3. Material and Methods

3.1. An overview on the dataset

The provided dataset contains 12,264 pairs, from which 10,424 (85%) are used for training and the rest for validation. Furthermore, the dataset comes with many peculiarities. A correlation between authors is challenging because the texts come from different writing scenarios. For example, most people follow a formal way of writing business memos. As a result, it is more difficult to find stylistic similarities between e-mails and business memos and text messages or essays, even if a pair is written by the same author.

When using transformers, the common practice is to encode the dataset in its original form without applying much preprocessing. After comparing emails and texts, we decided to remove HTML character artifacts as they do not contribute to authorship verification from a stylometric point of view. Since the Roberta model uses byte pair encoding, we suspected that the HTML characters would artificially increase a pair's dissimilarity.

3.1.1. Distribution of Text Length

First, an exploratory data analysis of the given dataset was conducted. The overall text length distribution at word level and the distribution of different types of text lengths is shown in Table 1 and Figure 1. This illustrates the necessity of assigning a high maximum length for tokenization, including padding and truncation.

The box plot shows that the length of the different discourse types varies greatly. The 'essay' type is much longer than the others, and the 'text message' type is the shortest.

	Overall	Essay	Text_Message	Email	Memo
mean	410	1718	96	1718	220
min	31	240	63	240	31
25%	96	1217	87	1217	169
50%	289	1603	93	1603	212
75%	363	2254	100	2254	292
max	3270	3270	474	3270	416

Table 1
Length of different types of text.

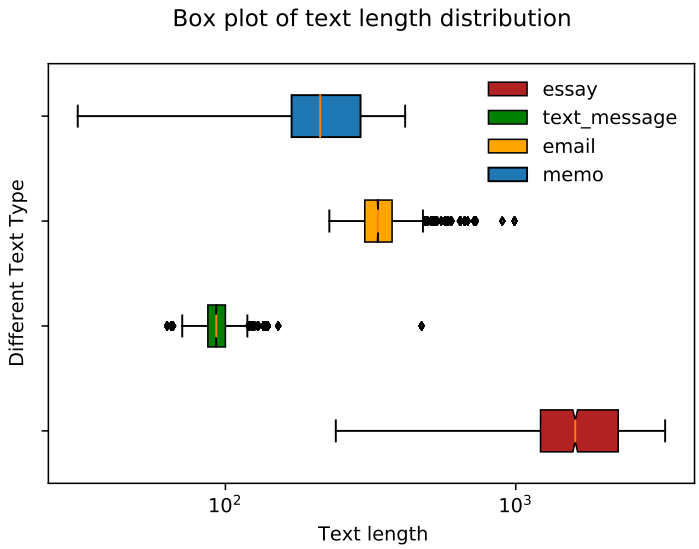

Figure 1: Box plot of the text length distribution of different text types. Granularity of counting is at the word level. Logarithmic scaling was applied to the x-axis.

3.1.2. Statistics for Pairs

To better understand the dataset, an analysis was performed at the discourse type (DT) level and the similarity was calculated for each of them – see table 2. Here, texts are encoded with the S-BERT model.

3.2. Encoder Experiments

The aim of our submission is to test pre-trained models of different architectures. These models are to be fine-tuned to this task using the dataset provided. For text comparison tasks such as semantic similarity or author verification, two approaches are popular: the Bi-Encoder and the Cross Encoder approach. These architectures are considered to be particularly powerful for similarity tasks.

MPNet [7] is trained using permuted language modeling and claims to gain a better understanding of bidirectional contexts, which may prove crucial for a text similarity task. The

Discourse Type Pairs (DT)	# Pairs	# Identical Authors	Mean Similarity
essay, email	1618	809	0.4850
email, text_message	7484	3742	0.6127
essay, text_message	1182	591	0.4803
memo, email	1014	507	0.4952
memo, text_message	780	390	0.5459
essay, memo	186	93	0.4161

Table 2
Analysis on each DT combination. 'Mean Similarity' is the mean of cosine similarity of all pairs in each DT combination.

Roberta models [8] that we used were already fine-tuned on similarity tasks, thus enabling the transfer of text similarity knowledge to our task. Therefore, we expect that the training time will be shorter and the performance will be better. The pretrained model configurations are shown in Table 3. The maximum length in Bi-Encoder models is 256, which is half of Cross-Encoder because Bi-Encoders encode two sentences separately: limiting the context to two 256 subtoken sequences, which are then concatenated. Note that for the experiments, only the pre-trained base models with a maximum text length of 512 subtokens were used.

Encoding Approach	Models	Max. Subtokens	Epochs	Batchsize
Cross Encoder	BertForNextSentencePrediction	512	8	16
Cross Encoder	Roberta-Muppet	512	5	6
Cross Encoder	Roberta-stsb	512	5	6
Bi-Encoder	MPNET	256	5	12
Bi-Encoder	Roberta-Muppet	256	5	12

Table 3
Details about the tested pretrained models. 'Max.Subtokens' is the value of hyperparameter 'max_length' in tokenizer. 'Batchsize' is the size of the batch used in training.

3.2.1. Cross-Encoder Approach

In the Cross-Encoder architecture, both texts are simultaneously fed through a transformer network. In this architecture, a single encoding for both texts is used for classification (see Figure 2 on the right). Cross-Encoders are normally used when you have a pre-defined set of text pairs that you want to score. Cross-encoders usually outperform Bi-Encoders, but do not scale well with large datasets. That's why Cross-Encoders seem suitable for our task.

For the Cross-Encoder approach, we used several Roberta models: Roberta-Muppet, Roberta fine-tuned on the STSB task [9], and plain BERT that was trained with the Next Sentence Prediction Task [10].

3.2.2. Bi-Encoder Approach

Bi-Encoder architecture creates a twin network that processes two sentences simultaneously in the same way [11]. All parameters are shared. The pooling layer creates fixed-size representation for input sentences of varying lengths, while also extracting the features that are considered the most important ones (see Figure 2 on the left).

For the Bi-Encoder approach, we experiment with Roberta-Muppet and MPNet adding a pooling and dense layer after the standard encoding step.

3.2.3. Cross-Encoder vs Bi-Encoder Approaches

The main difference between these two architectures is that two sentences are concatenated in a Cross-Encoder using the SEP special token as a separator. Thus, the encoder can have attention to information from both sentences, while in the Bi-Encoder architecture, the word embeddings of the two sentences are encoded separately and then concatenated. In addition, the Bi-Encoder does not compute attention information between the subtokens of the two texts, as the embedding process is done separately.

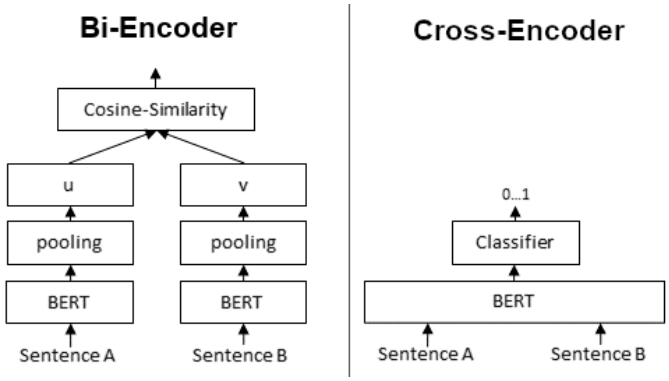

Figure 2: Schematic diagram of the structure of our two different encoder approaches [11]. Illustrated using a BERT model.

3.3. Dataset Manipulation Experiments

We also experimented with modifying and manipulating the dataset, in particular, splitting the text segments, negative sampling, and combining these two techniques. The purpose is to explore whether these techniques improve prediction results.

3.3.1. Splitting Text Segments

Splitting text segments in half results in twice as much data items as before. Since the models see smaller contexts during training, we hope that shorter but more texts force the model to better learn authorial characteristics that remain stable across various discourse types.

3.3.2. Negative Sampling

The negative sampling method was initially used to accelerate the training of Skip-Gram models [12] and has since been widely used in the natural language processing. Negative sampling serves two purposes: efficiency and effectiveness.

- **Efficiency**: Negative sampling can reduce the training load by optimizing only the vectors involved in the cost-finding process.
- **Effectiveness**: Negative sampling provides high-quality negative examples in a targeted manner, both to speed up convergence and allow the model to be optimized in the desired direction.

In this dataset experiment, all authors were mapped with their corresponding texts, and new pairs of negative data are created by combining texts from different authors.

Our aim was to create a dataset with 20% (6,132) positive samples and 80% (approx. 24,000) negative samples.

4. Evaluation

To evaluate the proposed models, we use the TIRA evaluation tool [13] with the following metrics:

AUC: The area-under-the-curve (ROC) score

F1-score: F1 score is the harmonic mean between precision and recall.

c@1: A variant of the F1-score, which rewards systems that leave difficult problems unanswered, like scores of exactly 0.5.

$\mathbf{F}_{0.5u}$: A measure that puts more emphasis on deciding same-author cases correctly.

Brier: The complement of the Brier score for evaluating the goodness of (binary) probabilistic classifiers.

The results of the models trained on this task will then be compared with a set of baseline results. The baseline results are obtained using a simple method that calculates the cosine similarities between TFIDF-normalized, bag-of-character-tetragrams representations of the text pairs. Then the resulting scores are shifted using a simple grid search, to arrive at an optimal performance on the validation set.

It has to be noted that the validation set across all models are always to be kept the same.

5. Results and Discussion

In Table 4, the results of the trained models are shown. Overall, the models scored decently on the validation dataset with an overall score mostly higher than 70%, higher than the baseline. MPNet using the Bi-Encoder approach scored the highest. Its permutation language modeling turned out to be better than the other models that used masked language modeling in pretraining.

The two Roberta-Muppet models performed similarly, falling right behind MPNet. At the same time, the two Roberta-Muppet models managed to perform considerably better than the

Model	auc	c@1	$F_{0.5u}$	F1	brier	Overall
Voting Ensemble 3 Models	0.765	0.759	0.718	0.800	0.759	0.760
Bi-MPNet	**0.777**	**0.771**	**0.729**	**0.807**	**0.771**	**0.771**
Bi-Roberta-Muppet	0.748	0.743	0.708	0.781	0.743	0.745
CE-Roberta-Muppet	0.749	0.744	0.709	0.782	0.744	0.746
CE-Roberta-stsb	0.68	0.672	0.65	0.745	0.672	0.684
CE-BertForNextSentencePrediction	0.705	0.701	0.679	0.724	0.701	0.702
Baseline	0.55	0.5	0.546	0.671	0.749	0.603
Bi-MPNet (on TIRA)	0.577	0.557	0.563	0.581	0.589	0.573

Table 4
Results of each model on the validation set and the best model's performance on the test set. "Bi-" prefix represents the Bi-Encoder Architecture & "CE-"prefix represents Cross-Encoder Architecture approach. The voting ensemble with 3 models includes: Bi-MPNet, Bi-Roberta-Muppet, CE-Roberta-Muppet. Bolded scores mark the best performance on each metric. Bi-MPNet (on TIRA) was evaluated on the test set of TIRA.

remaining two models. A possible explanation is that Roberta-Muppet's multitask pre-training translated into better stylometric understanding due to its more generalized embeddings.

In Figure 3, an analysis of the predictions of MPNet is shown to check the accuracy results on all combinations of discourse types on the validation set. The accuracy results range around 75% across all combinations. What is interesting is that all discourse type combinations achieved a similar score with marginal differences. Together with the $F_{0.5u}$ results, this indicates that up to a certain level, the model managed to learn authorial characteristics across discourse types.

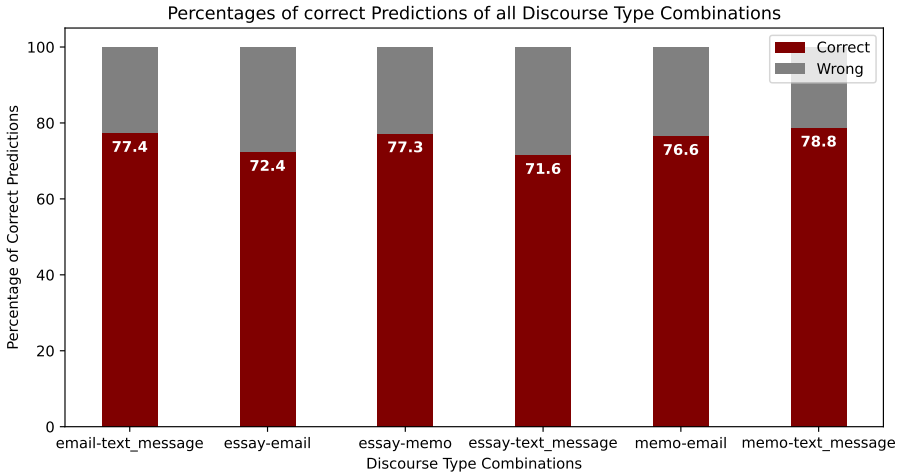

Figure 3: Accuracy results on all combinations of discourse types on the validation set.

Further experimentation is reported in Table 5. The models chosen for this analysis were the 3 best performing ones from Table 4. The results show that the dataset manipulation experiments

Model	auc			c@1			$F_{0.5u}$			F1			brier			overall		
	SPLIT	NEG	S.-N.	SPLIT	NEG	S.-N.	SPLIT	NEG	S.-N.	SPLIT	NEG	S.-N.	SPLIT	NEG	S.-N.	SPLIT	NEG	S.-N.
Bi-MPNet	0.689	0.499	0.501	0.689	0.499	0.501	0.674	0.555	0.556	0.739	0.666	0.667	0.689	0.499	0.501	0.696	0.544	0.545
Bi-Roberta-Muppet	0.640	0.501	0.672	0.640	0.501	0.672	0.637	0.556	0.658	0.691	0.667	0.742	0.640	0.501	0.672	0.650	0.545	0.684
CE-Roberta-Muppet	**0.729**	0.696	0.590	**0.729**	0.696	0.590	**0.705**	0.676	0.603	**0.770**	0.756	0.699	**0.729**	0.696	0.590	**0.732**	0.704	0.615

Table 5
Results of each model on the validation set on 3 experimental variants: "SPLIT" means splitting text segments in half (doubling the data item number). "NEG" is negative sampling. "S.-N." uses both SPLIT and NEG. "B-" prefix represents the Bi-Encoder approach and "CE-" prefix the Cross-Encoder. Underlined scores mark the experiment's (column) highest score and bold-ed scores mark the highest score of the metric.

did not contribute positively to the stylometric learning of the models. We observe a decrease in the performance of MPNet. Maybe splitting the data caused its permutation language model to be less effective given that less information for each shortened text, is encoded.

Negative sampling reduced the performance for all models. This discrepancy could be attributed to the positive to negative ratio of the dataset, therefore a smaller sample of negative data could have been better.

Finally, what we observe is that the Cross-Encoder models with Roberta-Muppet are clearly better than the other two models in the dataset manipulation experiments.

6. Conclusion

In our submission, we tested various pre-trained encoder models using Cross-Encoder and Bi-Encoder architectures to solve the authorship verification problem of PAN@CLEF 2022. We conclude that the inherent difficulty of the dataset is the major obstacle because the different types of discourse require different linguistic expressions, for instance, the considerable dissimilarity between business memos and text messages. The results show that a simple approach of selecting a pre-trained model and fine-tuning it is able to grasp some stylometric information useful for author verification, but overall the performance is not strong enough to reliably solve this difficult task.

Acknowledgments

Special thanks to Dr. Simon Clematide and Andrianos Michail for all the help and guidance given to our team for the completion of this work. Moreover, special thanks to the PAN members for the support given to us in situations of technical difficulties.

References

[1] Efstathios Stamatatos and Mike Kestemont and Krzysztof Kredens and Piotr Pezik and Annina Heini and Janek Bevendorff and Martin Potthast and Benno Stein, Overview of the Authorship Verification Task at PAN 2022, in: CLEF 2022 Labs and Workshops, Notebook Papers, CEUR Workshop Proceedings, 2022.

[2] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pezik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of PAN 2022: Authorship Verification, Profiling Irony and Stereotype Spreaders, and Style Change Detection, in: Alberto Barron-Cedeno, Giovanni Da San Martino, Mirko Degli Esposti, Fabrizio Sebastiani, Craig Macdonald, Gabriella Pasi, Allan Hanbury, Martin Potthast, Guglielmo Faggioli, Nicola Ferro (Ed.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), volume 13390 of *Lecture Notes in Computer Science*, Springer, 2022.

[3] M. Kestemont, E. Manjavacas, I. Markov, J. Bevendorff, M. Wiegmann, E. Stamatatos, B. Stein, M. Potthast, Overview of the cross-domain authorship verification task at pan 2021, in: CLEF (Working Notes), 2021, pp. 1743–1759. URL: http://ceur-ws.org/Vol-2936/paper-147.pdf.

[4] E. Stamatatos, Authorship verification: A review of recent advances, Research in Computing Science 123 (2016) 9–25. doi:10.13053/rcs-123-1-1.

[5] P. Juola, Authorship attribution, Foundations and Trends® in Information Retrieval 1 (2008) 233–334. doi:10.1561/1500000005.

[6] Z. Peng, L. Kong, Z. Zhang, Z. Han, X. Sun, Encoding text information by pre-trained model for authorship verification, in: CLEF, 2021.

[7] K. Song, X. Tan, T. Qin, J. Lu, T.-Y. Liu, Mpnet: Masked and permuted pre-training for language understanding, arXiv preprint arXiv:2004.09297 (2020).

[8] 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).

[9] P. May, Machine translated multilingual sts benchmark dataset., 2021. URL: https://github.com/PhilipMay/stsb-multi-mt.

[10] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[11] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).

[12] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems 26 (2013).

[13] M. Potthast, T. Gollub, M. Wiegmann, B. Stein, TIRA Integrated Research Architecture, in: N. Ferro, C. Peters (Eds.), Information Retrieval Evaluation in a Changing World, The Information Retrieval Series, Springer, Berlin Heidelberg New York, 2019. doi:10.1007/978-3-030-22948-1_5.