Improving Long-Text Authorship Verification via Model Selection and Data Tuning

Trang Nguyen **Kenneth Alperin** **Cagri Dagli** **Courtland VanDam** and **Elliot Singer**
MIT Lincoln Laboratory
Lexington, MA US
{trang.nguyen, kenneth.alperin, dagli, courtland.vandam, es}@ll.mit.edu

Abstract

Authorship verification is used to link texts written by the same author without needing a model per author, making it useful for de-anonymizing users spreading text with malicious intent. Recent advances in Transformer-based language models hold great promise for author verification, though short context lengths and non-diverse training regimes present challenges for their practical application. In this work, we investigate the effect of these challenges in the application of a Cross-Encoder Transformer-based author verification system under multiple conditions. We perform experiments with four Transformer backbones using differently tuned variants of fanfiction data and found that our BigBird pipeline outperformed Longformer, RoBERTa, and ELECTRA and performed competitively against the official top ranked system from the PAN evaluation. We also examined the effect of authors and fandoms not seen in training on model performance. Through this, we found fandom has the greatest influence on true trials, pairs of text written by the same author, and that a balanced training dataset in terms of class and fandom performed the most consistently.

1 Introduction

As more people turn to various online sources for their information, the ability to discriminate and discern authorship characteristics is critical to counter misinformation, plagiarism, and inappropriate aggregation. Understanding authorship is also essential to detecting individuals who make use of the anonymity afforded by the Internet to engage in harassment, impersonation, or criminal activities. Conversely, such technologies could also be applied in unethical ways such as the de-anonymization of whistle-blowers, for example. Additionally, the identification of bots and information operation campaigns is essential in the areas of cyber and national security.

Authorship analysis includes multiple tasks that address different use cases. The goal of author identification/attribution is to identify if a document was written by one of a known set of authors and, if yes, specify the individual. Author verification compares two documents to determine if they were written by the same author, without directly identifying or providing author information.

Advancements in authorship analysis have been furthered by efforts in the community. PAN[1] is a yearly series of shared tasks on important text forensics topics, including authorship. The PAN author verification task from 2020 (Bevendorff et al., 2020) and 2021 (Bevendorff and et al., 2021) uses a fanfiction dataset. Fanfiction has many interesting traits with respect to its use for automated authorship verification: the documents are long-text, authors can write stories in different fandoms (e.g., Harry Potter or Star Trek), and authors may emulate a certain style when writing within a specific fandom.

Traditional approaches to authorship recognition often focus on modeling lexical choice (i.e. word usage) or stylometry separately. Modern, deep learning-based approaches are much more expressive. New developments in Transformers and other large language models have been incredibly impactful in natural language processing and have been used to great success in tasks such as machine translation, text generation, and question and answering. Open-source communities make using these models easy and accessible.

For author recognition, these more expressive models have the potential to learn both lexical and stylometric information at the same time. One challenge in realizing this potential, however, is context length. Authorship is a subtle task and the more information the model can integrate at one time, the more of this subtlety can be captured. Further, diversity in training data can also play a

[1] https://pan.webis.de/

Proceedings of the 7th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL2023), pages 28–37

large role in the quality and robustness of trained models.

Accordingly, in this paper, we evaluate Transformer models for author verification on the long-text fanfiction data from PAN and attempt to understand the influence of topic and data tuning on performance. Our main contributions are the following:

- We evaluate four different Transformer models for author verification in terms of performance and their response to the fandom effect: standard models (RoBERTa, ELECTRA) and long-text models (Longformer, BigBird).
- We show that BigBird outperforms the other tested models and is competitive with systems submitted to PAN20/21.
- We demonstrate the impact of data tuning and preparation as an initial step into understanding how different aspects of a dataset influences model performance.

In the following sections, we first discuss work in the area of Transformers for authorship attribution and verification; outline the creation/tuning and statistical breakdown of our datasets; present our Cross-Encoder approach for verification; and then describe and discuss our experiments and results regarding the relative performance of multiple Transformer backbones, the fandom effect, and the performance of our BigBird Cross-Encoder on the official PAN20/21 test sets and how it is influenced by the dataset tuning.

2 Related Work

Previous approaches for author identification focused on traditional machine learning models with lexical information or stylometry, such as Burrow's Delta (Burrows, 2002). Deep learning approaches, like Transformers, have shown promising results for author identification. Fabien et al. (2020) developed a BERT approach and incorporated stylistic and hybrid features into their model to improve performance. Barlas and Stamatatos (2020) combined a multi-headed classifier (MHC) with pre-trained language models to evaluate their system's performance for cross-topic and cross-domain author verification (e.g., essays versus emails). They showed both the ELMo and BERT versions of their system outperformed a Recurrent Neural Network (RNN) baseline for cross-topic. Further in Barlas and Stamatatos (2021), they introduced transfer learning and evaluated an additional cross-fandom author identification scenario. This is different than our work, where we are using cross-fandom as our cross-topic scenario. In these experiments, their ELMo and ULMFiT systems outperformed their RNN baseline but was not SOTA for the target dataset.

Although pre-trained models overall appear promising for authorship analysis tasks, some work has been done that highlights possible limitations of these approaches. In Altakrori et al. (2021), the authors focused on the effect of topic and proposed a topic confusion task, where author and topic pairs are swapped between the train and test datasets.

Transformers' use in author verification has mixed results. Manolache et al. (2021) evaluated several BERT-like models for author verification using the PAN20/21 data with good success. However, their experiments also indicated these models relied on topical information rather than authorship characteristics. As in our work, the authors investigated how data partitioning affected model performance. However, this work was limited to dataset tuning based on disjoint authors or fandoms.

Ordoñez et al. (2020) used Longformer for the PAN20 challenge but had very different results on their test splits and the official PAN test set. Their model performed worse than the baselines provided by PAN. Conversely, in Peng et al. (2021), the authors used a BERT-based model for PAN21 (open-set scenario) and had promising results when compared to other models trained on the small dataset.

These works inspired us to explore how pre-trained Transformer models performed for author verification. PAN20/21 is a great source of long-text data, so we compared general Transformers with ones specialized for long-text. We also studied how fandoms and datasets affect performance.

3 Datasets

PAN offered data for closed and open-set author verification tasks. We used four training/validation sets and four test sets, with all training sets and two test sets derived from the PAN training data.

3.1 PAN20/21 Official Data Overview

The data used came from the PAN20 and PAN21 authorship verification tasks, which provided an official training dataset (with small/large versions) and two official test sets for the closed-set/open-

set cases. These datasets consist of fanfiction text trial pairs. We used the large training set (PAN20 Shuffled), with 490k texts by 278k authors in 1.6k fandoms. The PAN test sets are smaller at 28.6k texts/12.6k authors and 49k texts/40k authors respectively. Both contain 400 fandoms. For each trial, PAN provides the fandoms and raw texts. Texts can appear in multiple trial pairs. In order to be as generalizable as possible, our models did not use the fandom information.

3.2 PAN20 Curated Datasets

To better study the effect of topic as a confounder, we resampled pairs from the official PAN20 training corpus to create new sets of splits for closed- and open-set verification conditions which we refer to as PAN20 Curated (Closed) and PAN20 Curated (Open), respectively. These datasets were created without prior knowledge about the structure of the official test datasets.

3.2.1 Curation and Post-Processing

We first separated the given trial pairs in the PAN20 large training set into texts by author and assigned unique story IDs to texts to create a pool of stories for resampling. We removed "inactive" authors with fewer than 20 associated texts in the corpus.

To investigate the role of topical variation, we designed splits to assess systems' abilities to model authorship within/across fandoms by bi-clustering stories based on authorship and fandom. We first formed the active-author-fandom matrix, and then performed spectral co-clustering to create four unique author-fandom co-clusters. Each quadrant (00, 01, 10, and 11) represents a unique grouping of stories with respect to author and fandom. The main clusters, 00 and 11, are completely disjoint in terms of authors and fandoms. Diagonal clusters, 01 and 10, contain the subset of texts that overlap in author/fandom of the main clusters.

For the closed-set verification scenario, PAN20 Curated (Closed), the training data consists of stories from the 00 and 11 author-fandom cluster conditions. Validation and test data are sampled from clusters 01 and 10. For the open-set scenario, PAN20 Curated (Open), the training data consists of story pairs sampled uniformly from the 00 author-fandom condition.

To reduce biases in the validation and test datasets, we did post-filtering to rebalance the number of authors and fandoms. We then sampled trial pairs from each cluster. Validation and test datasets are sampled from the filtered set of stories. We sampled uniformly across combinations of fandoms within trial pairs and set the open-set condition at 60% of all pairs.

3.3 PAN20 Equal Dataset

We created the PAN20 Equal training dataset to have an equal number of trials of each type: same author within the same fandom (TT WIN), same author between fandoms (TT BW), different authors within the same fandom (FT WIN), and different authors between fandoms (FT BW). Authors and their unique texts were randomly sampled and recombined to create trials for each type. The total number of trials per type was arbitrarily capped.

3.4 Statistical Breakdown of Datasets Used

To investigate how dataset tuning and features affect performance, we tabulated the trials, authors, and fandoms represented over each trial type. These can be seen in Table 1. We defined trials by two characteristics: whether the trial was a TT or FT pair, and whether the trial text was WIN or BW fandoms. The tables show the unique number of trials, authors, and fandoms for that trial type. The PAN20 Curated datasets (Closed and Open) have the most trials in both train and test, with PAN20 Curated Closed having the most with 780k/210k train/test. Most of the datasets have a smaller proportion of TT WIN trials, and the official PAN20/21 Test data sets and PAN20 Shuffled have none of this trial type. These three datasets also have a relatively small percentage of FT WIN.

Author distribution is fairly equal across the trial types for PAN20 Curated (Closed and Open) because this was a tuning focus. However, they contain few authors relative to the other datasets. PAN20 Shuffled contains the most unique authors at 227k total. The representation of authors in PAN20 Shuffled, PAN20 Equal, and the PAN20/21 Test data sets is skewed towards FT BW, as the data has a large number of single text authors and fandoms with few texts.

In terms of fandom distribution, all the training datasets contained a majority of the available 1600 fandoms, except for PAN20 Curated (Open) (with only 784) because of its post-processing. Similarly, the PAN20 Curated (Open) Test set also contained the fewest unique fandoms at 204, while PAN20 Curated (Closed) Test contained more than double the official test sets with 1151 fandoms. FT WIN trials had the least fandom representation, except

	PAN20 Curated (Closed)		PAN20 Curated (Open)		PAN20 Shuffled		PAN20 Equal	
Train Dataset Statistics	**TT**	**FT**	**TT**	**FT**	**TT**	**FT**	**TT**	**FT**
Trial Pairs WIN	18869	186885	10679	106640	0	18388	39538	39538
Trial Pairs BW	90179	481572	51204	241032	118392	83672	39538	39538
Total Trial Pairs	777505		409555		220452		158152	
Authors WIN	2584	2590	1446	1452	0	36776	14402	31531
Authors BW	2590	2592	1452	1452	36591	165045	18955	56560
Total Authors	2592		1452		227274		60366	
Fandoms WIN	1251	759	703	408	0	252	1525	1522
Fandoms BW	1383	1393	773	784	1600	1600	1593	1589
Total Fandoms	1393		784		1600		1597	
Test Dataset Statistics	**TT**	**FT**	**TT**	**FT**	**TT**	**FT**	**TT**	**FT**
Trial Pairs WIN	8184	49580	3760	6099	0	209	0	992
Trial Pairs BW	31404	120994	5572	10942	7786	6316	10000	9007
Total Trial Pairs	210162		26373		14311		19999	
Authors WIN	1594	1587	280	269	0	418	0	1984
Authors BW	1456	1615	249	280	2907	11139	7615	18014
Total Authors	1615		280		12636		27613	
Fandoms WIN	1044	531	200	115	0	5	0	20
Fandoms BW	1140	1151	196	198	399	400	400	388
Total Fandoms	1151		204		400		400	

Table 1: Unique trial, author, and fandom counts for train and test datasets

in PAN20 Equal where each trial type had roughly the same number of unique fandoms.

Our datasets had differences in the extent and focus of their tuning as shown by the trial type, author, and fandom distributions. This variation in datasets allowed us to more thoroughly evaluate our system approach and Transformer backbones.

4 System Approach

We proposed a Transformer-based Cross-Encoder model setup for authorship verification that allowed us to evaluate several Transformer backbones and compared them to the baseline from PAN. This baseline (called "naïve" in the PAN official results and "cosine" in ours) is based on Term Frequency-Inverse Document Frequency (TF-IDF) cosine similarity computed over word tokens.

4.1 Cross-Encoder Model

Our Cross-Encoder system was designed to use existing pre-trained models from HuggingFace (Wolf et al., 2020). With a cross-encoder, each trial pair is passed to the classifier without creating individual text embeddings.

Training and validation trial pairs are subsampled in a balanced fashion with respect to TT/FT. The exact number of pairs used for train/validation is specified during experiment setup. We evaluated the impact of sample size on performance but only show results for one subsample in this paper. Text pairs are tokenized using the associated Huggingface Transformer tokenizer then passed to the Transformer backbone for classification.

We also included an option for "windowing" trials prior to tokenization. When windowing, a window equal to half the maximum length (dependent on the specified token limit) is randomly chosen for each text in the pair. We predicted windowing would improve performance, particularly when using smaller token limits, since the window of text can be pulled from any part of the story and different windows of the same story are used over multiple epochs thereby increasing coverage. At inference time, scores from multiple windowings of a test pair are pooled and returned as the final test pair score.

4.2 Transformer Backbone

We performed experiments with four Transformer backbones. DistilRoBERTa and ELECTRA have a token limit of 512 but use different pre-training approaches. DistilRoBERTa is the distilled version

Model	Windowing	Learning Rate	Gradient Clip	Precision	Batch Size
BigBird	Y	3.00E-03	0	16	2
Longformer	N	5.00E-04	0	16	4
DistilRoBERTa	Y	3.00E-04	1	32	16
ELECTRA	Y	3.00E-04	0	16	4

Table 2: Optimal hyper-parameters for each transformer backbone

	Model	PAN20 Curated (Open) Test		PAN20 Test		PAN21 Test	
		EER	AUC	EER	AUC	EER	AUC
PAN20 Curated (Open) Train	**BigBird**	**0.067**	**0.982**	**0.08**	**0.976**	**0.081**	**0.975**
	Longformer	0.221	0.869	0.224	0.855	0.251	0.831
	DistilRoBERTa	0.192	0.893	0.226	0.856	0.192	0.889
	ELECTRA	0.261	0.815	0.326	0.739	0.311	0.754
	Cosine Baseline	0.235	0.841	0.293	0.778	0.274	0.797
PAN20Shuffled Train	**BigBird**	**0.082**	**0.976**	**0.072**	**0.979**	**0.048**	**0.990**
	Longformer	0.143	0.936	0.144	0.928	0.109	0.959
	DistilRoBERTa	0.258	0.818	0.230	0.853	0.172	0.907
	ELECTRA	0.270	0.813	0.221	0.862	0.178	0.904
	Cosine Baseline	0.237	0.838	0.297	0.780	0.281	0.798

Table 3: Results for Transformer-backbone Cross-Encoder Models for two training sets and three test sets

of RoBERTa (a model that builds and improves on the original BERT model) and uses Masked language modeling (MLM) and next sentence prediction (Sanh et al., 2019). ELECTRA uses the same underlying BERT model but is pre-trained on a task called replaced token detection (RTD), which was shown to be more efficient for some problem sets (Clark et al., 2020). We show the results from ELECTRA Large.

Longformer and BigBird are Transformers designed for longer text and have a token limit of 4096. Both are based on RoBERTa. Longformer's approach to self-attention is to use global attention and a sliding window for local context (Beltagy et al., 2020). Although Longformer can notionally have dilated windows, the HuggingFace implementation does not support this option. BigBird has a slightly different approach to self-attention, and uses a combination of global attention, windows for local context, and random attention (Zaheer et al., 2020).

5 Experimental Results and Discussion

Our approach was to evaluate our Cross-Encoder model using multiple Transformer backbones on datasets with different types of tuning, and then compare its performance to the PAN baseline systems. We first optimized the hyper-parameters for each backbone, then examined the model's performance and effect of fandom.

After identifying BigBird as the backbone with the best performance, we evaluated the Cross-Encoder model using the metrics from the PAN challenge across all combinations of the multiple dataset variants.

5.1 Setup and Hyper-Parameter Selection

We conducted all Cross-Encoder experiments by subsampling to 50k pairs for training and 2k pairs for validation. Using larger subsamples did not dramatically increase performance. Each Transformer used its maximum token limit. We used twenty epochs for training, with early stopping after three epochs of no improvement. We scored each test trial using five different window-pairs and average-pooling to report the final score for each test trial.

We ran experiments with the Cross-Encoder models and a range of hyper-parameters to identify the optimal hyper-parameters for each Transformer backbone. Hyper-parameters that differed among Cross-Encoder models are shown in Table 2.

The learning-rate, gradient clipping, batch size, and windowing all had significant impact on the system performance. The precision did not affect performance, but it along with the batch size were limited by the hardware available.

We found only the Longformer Cross-Encoder

did not improve with windowing, while only the DistilRoBERTa Cross-Encoder benefited from gradient clipping. The BigBird Cross-Encoder did best with a larger learning rate but required a smaller batch size.

5.2 Comparison of Transformer-Based Cross-Encoders

The area under the curve (AUC) and equal error rate (EER) of the Cross-Encoder models and cosine baseline for two training sets and three test sets are shown in Table 3. Note we do not show all the dataset combinations here for simplicity.

The BigBird Cross-Encoder model outperforms regardless of train and test dataset, while the ELECTRA backbone tends to have poor performance. All Transformer Cross-Encoders performed best in the PAN20 Shuffled train/PAN21-Test experiment, with EER ranging from 4.8% for BigBird to 17.8% for ELECTRA.

As shown in the detection error tradeoff (DET) plots in Figure 1, the training data used impacts relative performance of the DistilRoBERTa and Longformer Cross-Encoders. For PAN20 Curated (Open) train/PAN21-Test, DistilRoBERTa outperforms Longformer, which is unexpected given that Longformer is meant for long text. However, when trained with PAN20 Shuffled, results are as expected: the long-text-specific Longformer does better than the more general DistilRoBERTa.

The Longformer backbone's relative performance inconsistency appears due to sensitivity to the training dataset. The Longformer Cross-Encoder EER increased from 10.9% to 25.1% when training with PAN20 Shuffled versus PAN20 Curated Open for the PAN21-Test experiment. The ELECTRA Cross-Encoder model has a similar sensitivity, and its EER increased from 17.8% to 31.1%. Comparatively, the DistilRoBERTa system was more stable (17.2% –> 19.2%).

5.3 Fandom Effect

To further explore the effect of topic, we considered fandom match/mis-match at the pair level (i.e., between TTs and FTs). These results are shown in Table 4, again for PAN20 Shuffled Open train/test. Note that ELECTRA and Longformer Cross-Encoder results are not shown but are consistent with DistilRoBERTa. Systems show a similar pattern, with fandom appearing to have a particularly strong influence on performance of TT pairs. For the BigBird Cross-Encoder, the highest performing breakout experiment (TTs from within the same fandom, FTs from between fandoms) has an EER of 0.98%, which is much lower than the average EER of approximately 6.7%. This may be because for this condition, the system can lean on its ability to match similar topical content (within fandom TTs) and discriminate between different topical content (between fandom FTs). The breakout condition where this ability is not as useful is the lowest performing breakout condition (TTs from between fandoms, FTs from within the same fandom), where the performance is an order of magnitude worse (nearly 10% EER). In this case, matching/discriminating topical content is actually a hindrance to performance. A primary difference between these results and those of the other Transformer systems is that the BigBird Cross-Encoder has effectively eliminated the effect of within/between fandom for FTs. We focus on the BigBird Cross-Encoder system going forward due to its strong performance.

5.4 Datasets Comparison using BigBird Cross-Encoder

Table 5 shows the performance of our BigBird Cross-Encoder model and the cosine baseline for multiple combinations of the four training and four test datasets. This table includes two performance metrics for each experiment: AUC and the official PAN challenge score (the average of the AUC, F1, F0.5u, c@1, and Brier score). These scores were calculated using the offical PAN scoring code.[2]

Because PAN introduced the notion of "unanswered" trials in the challenge and scoring, we included two versions of our BigBird model: the original version and a modified version that manually sets scores that round to 0.5 to "unanswered" (denoted by *). This was done to evaluate the uncertainty of our system. Our original Cross-Encoder system does not leave trials unanswered, so we created the BigBird Cross-Encoder* to naively allow it to mark difficult trials.

The BigBird Cross-Encoder model did well for all dataset combinations and significantly outperformed the cosine baseline. Naively leaving trials unanswered with BigBird Cross-Encoder* generally increased the overall PAN score. BigBird Cross-Encoder* assigned less than 2% of total trials

[2] https://github.com/pan-webis-de/pan-code/tree/master/clef22/authorship-verification

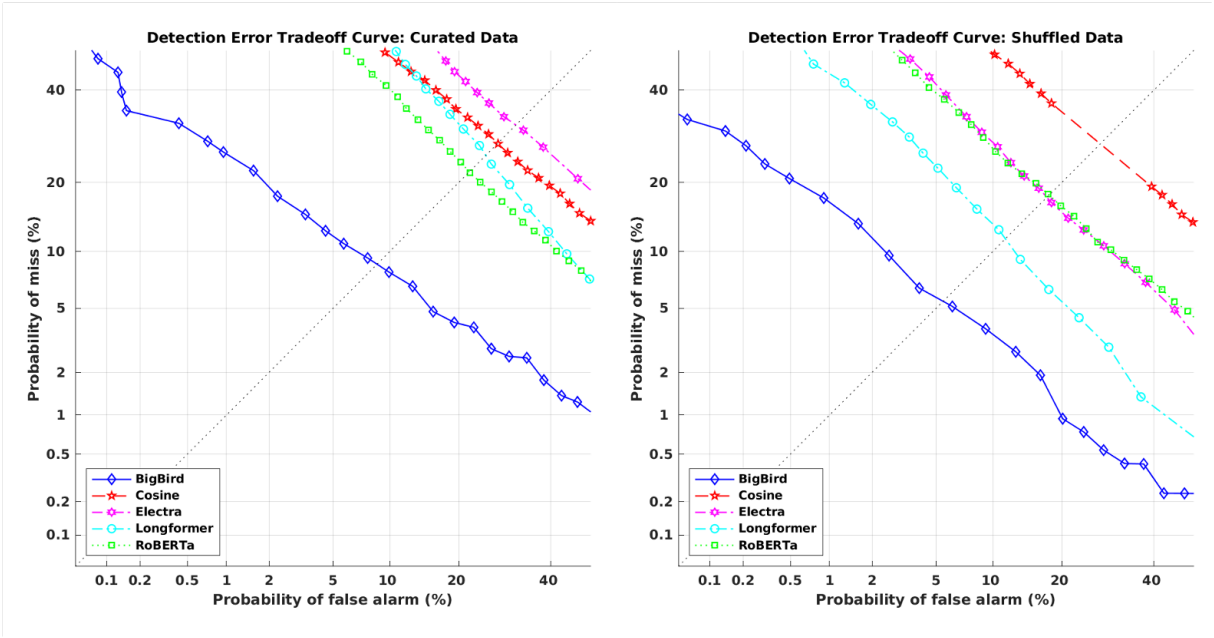

Figure 1: DET plots of Transformer-Based Cross-Encoder performance comparison for PAN21-Test trained using PAN20 Curated (Open) on the left and PAN20 Shuffled on the right. (Lines closer to the lower left are better)

			False Trials (FT)			
			Within Fandom		Between Fandom	
			AUC	EER	AUC	EER
DistilRoBERTa	**True Trial (TT)**	**Within Fandom**	0.981 ± 0.002	0.068 ± 0.006	0.994 ± 0.001	0.035 ± 0.005
		Between Fandom	0.783 ± 0.009	0.288 ± 0.009	0.864 ± 0.008	0.220 ± 0.010
BigBird	**True Trial (TT)**	**Within Fandom**	0.999 ± 0.001	0.009 ± 0.007	0.999 ± 0.001	0.0098 ± 0.006
		Between Fandom	0.967 ± 0.008	0.091 ± 0.016	0.966 ± 0.008	0.088 ± 0.015

Table 4: Performance breakdown to show fandom effect by trial type using PAN20 Curated (Open)

	Model	PAN20 Curated (Closed) Test		PAN20 Curated (Open) Test		PAN20 Test		PAN21 Test	
		AUC	PAN	AUC	PAN	AUC	PAN	AUC	PAN
PAN20 Curated (Closed) Train	**BigBird**	0.987	0.9215	-	-	0.980	0.9469	0.977	0.9381
	BigBird*	0.987	0.9268	-	-	0.980	0.9498	0.977	0.9421
	Cosine Baseline	0.805	0.7098	-	-	0.779	0.5635	0.798	0.6110
PAN20 Curated (Open) Train	**BigBird**	-	-	0.982	0.9416	0.976	0.9262	0.975	0.9352
	BigBird*	-	-	0.982	0.9452	0.976	0.9292	0.975	0.9384
	Cosine Baseline	-	-	0.841	0.7840	0.778	0.4474	0.797	0.6416
PAN20 Shuffled Train	**BigBird**	0.975	0.8976	0.976	0.9224	0.979	0.9416	0.990	0.9582
	BigBird*	0.975	0.9044	0.976	0.9264	0.979	0.9440	0.990	0.9596
	Cosine Baseline	0.806	0.5893	0.838	0.7212	0.780	0.7554	0.798	0.7610
PAN20 Equal Train	**BigBird**	0.983	0.9280	0.981	0.9426	0.982	0.9370	0.984	0.9416
	BigBird*	0.983	0.9325	0.981	0.9454	0.982	0.9390	0.984	0.9436
	Cosine Baseline	0.805	0.7020	0.837	0.7874	0.780	0.5234	0.797	0.7048

Table 5: BigBird Cross-Encoder performance for various dataset combinations. Scores in grey are best across systems

unanswered for all experiments, which was fewer than the cosine baseline (4.37% to 20.15% unanswered). This indicates the BigBird Cross-Encoder model has more separation in its TT and FT predictions than the baseline model.

No training set performed best across all test datasets. However, we did notice some patterns in the results. First, training datasets generally performed best with test sets that matched in terms of distribution of trials, authors, and fandoms, e.g., PAN20 Curated performed the best with the PAN20 Curated test sets, and PAN20 Shuffled performed best with PAN21-Test. While this does not hold for the case of PAN20-Test, this is a more complicated comparison because the results are a mixture of open- and closed-set verification.

The second observation is PAN20 Equal performed consistently for all test sets, even though it contains the fewest total trials and has fewer authors than PAN20 Shuffled. This could be a first step towards identifying a tuning approach for generalizable datasets. Although systems trained with this dataset do not always achieve top performance, they do outperform compared to other training sets in at least some of the individual performance metrics for all test sets except PAN21-Test. It is unlikely that the distribution of trials, authors, and/or fandoms in a test set of interest will always be known, so understanding what makes a training dataset more general is critical.

For the official PAN20-Test and PAN21-Test datasets, the best training datasets were PAN20 Curated (Closed) and PAN20 Shuffled respectively. Table 6 shows the BigBird Cross-Encoder performance using these training datasets compared to the official results of the PAN20/21 challenge top participant systems (Bevendorff and et al., 2021). These include hybrid neural-probabilistic, neural network-based, logistic regression, and graph-based Siamese network systems (Boenninghoff et al., 2020, 2021; Weerasinghe and Greenstadt, 2020; Embarcadero-Ruiz et al., 2021). Note here the systems submitted by the same team are not necessarily the same across PAN20 and PAN21 because some systems used for the PAN20 closed-set challenge relied on fandom information. The BigBird Cross-Encoder* model performed competitively with the top performers from the challenge, and can be used without modification for both tasks since it does not use fandom data. While this table shows our best results, the PAN score was > 0.9 for every training dataset we evaluated. Overall, for the PAN20/21 challenge the BigBird Cross-Encoder model performed very well, despite having a straightforward architecture and using a naive approach to leaving trials unanswered. There was limited benefit in using the tuned training datasets for PAN, potentially because the provided official training data matched distributions of the official test data so well. Future work will entail leveraging explainable AI techniques to understand black-box aspects of these models, including why BigBird is less affected by variations in training regminens.

6 Conclusion

We compared several Transformer backbones with our Cross-Encoder systems and found the choice in backbone dramatically impacted the feasibility of our Cross-Encoder model for long-text authorship verification. BigBird outperformed another long-text Transformer (Longformer) and two general Transformers that use different pre-training approaches (DistilRoBERTa and ELECTRA). Our experiments show that Longformer and ELECTRA are both sensitive to the tuning and preparation of training data. Our Longformer results were consistent with Ordoñez, et. al (2020); this sensitivity to datasets makes Longformer and ELECTRA non-ideal candidates for this task.

We found that fandom (which we considered equivalent to topic) is particularly important for TTs. TTs that are between fandom were significantly more difficult for our system to correctly predict than those that were within fandom. This fandom effect was seen to a lesser extent for FTs but was eliminated in BigBird Cross-Encoder. This visible fandom effect indicates that there is still room for future work to improve the model's ability to learn features of the author and reduce reliance on fandom.

Our BigBird Cross-Encoder performed very competitively with the official PAN20/21 scores and outperformed the top system for both the closed- and open-set verification tasks with only a naïve approach to leaving hard trials unanswered. These results show that BigBird may have great potential for author recognition work.

The BigBird Cross-Encoder performed well on PAN20/21 test sets without extra tuning, but different data tuning approaches affect system performance on test sets. For example, the minimally processed PAN20 Shuffled did not work the best for

	Team	Training	AUC	F1-Score	F0.5u	c@1	Brier	Overall
PAN20	boenninghoff20	large	0.969	0.936	0.907	0.928	-	0.935
	weerasinghe20	large	0.953	0.891	0.882	0.88	-	0.902
	boenninghoff20	small	0.94	0.906	0.853	0.889	-	0.897
	weerasinghe20	small	0.939	0.86	0.817	0.833	-	0.862
	BigBird Cross-Encoder*	PAN20 Curated (Closed)	0.980	0.938	0.947	0.934	0.946	0.950
PAN21	boenninghoff21	large	0.9869	0.9524	0.9378	0.9502	0.9452	0.9545
	embarcaderoruiz21	large	0.9697	0.9342	0.9147	0.9306	0.9305	0.9359
	weerasinghe21	large	0.9719	0.9159	0.9245	0.9172	0.9340	0.9327
	weerasinghe21	small	0.9666	0.9071	0.9270	0.9103	0.9290	0.9280
	BigBird Cross-Encoder*	PAN20 Shuffled	0.9900	0.9440	0.9620	0.9460	0.9560	0.9596

Table 6: Comparison of BigBird Cross-Encoder and PAN top performing systems

the PAN20 Curated test sets. Matching the distribution of trials, authors, and fandoms between train and test data led to the best performance, but this approach is not necessarily feasible for real-world applications. We found that the PAN20 Equal training data, which was tuned for equal trial types, performed consistently across all the test sets. More research is needed to determine what aspects of this tuning actually affects performance, and if PAN20 Equal is also generalizable to other test sets or the approach to other types of data.

Limitations

For our Cross-Encoder systems, the Transformer backbones we evaluated vary in their memory and GPU requirements, but the best performing backbone (BigBird) has greater hardware needs than may be available to some researchers. Similarly, BigBird requires more time for training and testing and could take multiple days to train. We also found that 50k trials were sufficient for training, but this amount of training data may not be available for all use cases.

Our experiments and findings focused on fandoms (or topics) and data tuning could be difficult to evaluate on other datasets because of the additional requirement for topic labels, which may not be found in all author attribution datasets. Depending on the data source, some documents may also have multiple topic labels, which is not considered in our work.

Ethics Statement

While there are many legitimate use cases for authorship analysis, it is also possible to use these approaches in a way that negatively impacts people's freedom, livelihood, or safety. For example, these models could be used to de-anonymize texts written by whistle-blowers, protesters, or other dissidents. People may also face personal embarrassment, social stigma, or loss of employment if they are linked with texts shared under the assumption of anonymity.

Acknowledgements

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited. This material is based upon work supported under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Air Force. Delivered to the U.S. Government with Unlimited Rights, as defined in DFARS Part 252.227-7013 or 7014 (Feb 2014). Notwithstanding any copyright notice, U.S. Government rights in this work are defined by DFARS 252.227-7013 or DFARS 252.227-7014 as detailed above. Use of this work other than as specifically authorized by the U.S. Government may violate any copyrights that exist in this work.

References

Malik Altakrori, Jackie Chi Kit Cheung, and Benjamin CM Fung. 2021. The topic confusion task: A novel evaluation scenario for authorship attribution. In *Findings of the Association for Computational Linguistics: EMNLP 202*, pages 4242–4256.

Georgios Barlas and Efstathios Stamatatos. 2020. Cross-domain authorship attribution using pre-trained language models. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 255–266.

Georgios Barlas and Efstathios Stamatatos. 2021. A transfer learning approach to cross-domain authorship attribution. *Evolving Systems*, 12(3):625–643.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020.

Longformer: The long-document transformer. *Computing Research Repository*, arXiv:2004.05150.

Janek Bevendorff and et al. 2021. Overview of pan 2021: Authorship verification, profiling hate speech spreaders on twitter, and style change detection. In *Advances in Information Retrieval*, pages 567–573.

Janek Bevendorff, Bilal Ghanem, Anastasia Giachanou, Mike Kestemont, Enrique Manjavacas, Ilia Markov, Maximilian Mayerl, Martin Potthast, Francisco Rangel, Paolo Rosso, et al. 2020. Overview of pan 2020: Authorship verification, celebrity profiling, profiling fake news spreaders on twitter, and style change detection. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11*, pages 372–383. Springer.

Benedikt Boenninghoff, Robert M. Nickel, and Dorothea Kolossa. 2021. O2d2: Out-of-distribution detector to capture undecidable trials in authorship verification. *PAN@CLEF 2021*, arXiv:2106.15825.

Benedikt Boenninghoff, Julian Rupp, Robert M. Nickel, and Dorothea Kolossa. 2020. Deep bayes factor scoring for authorship verification. *CLEF 2020 Labs and Workshops*, arXiv:2008.10105.

John Burrows. 2002. 'delta': a measure of stylistic difference and a guide to likely authorship. *Literary and linguistic computing*, 17(3):267–287.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *Computing Research Repository*, arXiv:2003.10555.

Daniel Embarcadero-Ruiz, Helena Gomez-Adorno, Ivan Reyes-Hernandez, Alexis Garcia, and Alberto Embarcadero-Ruiz. 2021. Graph-based siamese network for authorship verification. In *Notebook for PAN at CLEF 2021*.

Maël Fabien, Esaú Villatoro-Tello, Petr Motlicek, and Shantipriya Parida. 2020. Bertaa: Bert fine-tuning for authorship attribution. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 127–137.

Andrei Manolache, Florin Brad, Elena Burceanu, Antonio Barbalau, Radu Ionescu, and Marius Popescu. 2021. Transferring bert-like transformers' knowledge for authorship verifications. *Computing Research Repository*, arXiv:2112.05125.

Juanita Ordoñez, Rafael A. Rivera Soto, and Barry Y. Chen. 2020. Will longformers pan out for authorship verification? In *Notebook for PAN at CLEF 2020*.

Zeyang Peng, Leilei Kong, Zhijie Zhang1, Zhongyuan Han, and Xu Sun. 2021. Encoding text information by pre-trained model for authorship verification. In *Notebook for PAN at CLEF 2021*, pages 1–5.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Computing Research Repository*, arXiv:1910.01108.

Janith Weerasinghe and Rachel Greenstadt. 2020. Feature vector difference based neural network and logistic regression models for authorship verification. In *Notebook for PAN at CLEF 2020*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. *Computing Research Repository*, arXiv:2007.14062.