Overview of the Cross-Domain Authorship Verification Task at PAN 2020

Mike Kestemont,[1] Enrique Manjavacas,[1] Ilia Markov,[1] Janek Bevendorff,[2]
Matti Wiegmann,[2] Efstathios Stamatatos,[3] Martin Potthast,[4] and Benno Stein[2]

[1]University of Antwerp
[2]Bauhaus-Universität Weimar
[3]University of the Aegean
[4]Leipzig University

pan@webis.de https://pan.webis.de

Abstract Authorship identification remains a highly topical research problem in computational text analysis with many relevant applications in contemporary society and industry. For this edition of PAN, we focused on authorship verification, where the task is to assess whether a pair of documents has been authored by the same individual. Like in previous editions, we continued to work with (English-language) fanfiction, written by non-professional authors. As a novelty, we substantially increased the size of the provided dataset to enable more data-hungry approaches. In total, thirteen systems (from ten participating teams) have been submitted, which are substantially more diverse than the submissions from previous years. We provide a detailed comparison of these approaches and two generic baselines. Our findings suggest that the increased scale of the training data boosts the state of the art in the field, but we also confirm the conventional issue that the field struggles with an overreliance on topic-related information.

1 Introduction

From the very beginning, authorship analysis tasks have played a key role within the PAN series. A variety of shared tasks have been developed over the past decade, complemented by the much-needed development of benchmark corpora for problems such as authorship attribution, authorship clustering, and authorship verification – both within and across genres, and within and across languages. Rather than adding new task variants (or repeating existing ones), we decided this year to renew our mission and broaden our perspective, by organizing an annual series of tasks of a gradually increasing difficulty and realism, organized within a three-year strategy (2020-2023). In this endeavour, we also aim to integrate as many of the lessons learned from recent editions as possible. Amongst others, we aim to devote explicit care to some of the larger challenges that remain open in the field, such as author-topic orthogonality, cross-genre

issues and problems involving texts of unequal lengths. Additionally, based on the corpus construction efforts of Bischoff et al. [6], we can provide evaluation data of a more substantial size than in previous years, so as to keep the field up to speed with broader developments in NLP, and especially the emergence of data-hungry methods from representation learning. This endeavour puts demanding constraints on the practical organization of the annual task, but it is our hope that this renewed strategy will be beneficial to participants and the whole field in general.

The task in year one, which is this year, was deliberately formulated as a closed setting: The test set only contains a subset of authors and topics that were also present in the training set. While presented as an authorship verification task, it could therefore also be seen as an authorship attribution task. The task in year two will move beyond this and is designed in a much more demanding open setting. While the training set in year two will be identical to the one from this year, the test set will no longer contain any authors or topics that were also present in the training set. This setup entails a highly challenging task in (pure) authorship verification. The task in year three (currently termed as "surprise task") is intended to put the participants in the role of judges at court, where the highest possible reliability and dependability will be required. These requirements will be reflected in the evaluation measures. More details on this task will be released in due time.

2 Authorship verification

The automated authentication of textual documents is an important issue in computer science, with multiple, real-world applications across various domains. Trustworthy, transparent benchmark initiatives are therefore key in reliably establishing the state of the art in the field, stimulating replication, and monitoring progress [27]. Automatically assessing a document's authorship on the basis of its linguistic and stylistic features is a crucial aspect of this problem [30, 17, 22], which can be modeled in various ways. Over the years, PAN has contributed to the field of authorship identification through organizing an array of shared tasks that were recently surveyed [3]. Some of the core tasks included:

- AUTHORSHIP ATTRIBUTION: given a document and a set of candidate authors, determine which of them wrote the document (2011-2012, 2016-2020);
- AUTHORSHIP VERIFICATION: given a pair of documents, determine whether they are written by the same author (2013-2015);
- AUTHORSHIP OBFUSCATION: given a document and a set of documents from the same author, paraphrase the former so that its author cannot be identified anymore (2016-2018);
- OBFUSCATION EVALUATION: devise and implement performance measures that quantify safeness, soundness, and/or sensibleness of an obfuscation software (2016-2018).

As previously announced [3], we have revisited the task of authorship verification this year, which can be formalized as the task of approximating the target function $\phi : (D_k, d_u) \rightarrow \{T, F\}$, where D_k is a set of k documents of known authorship

by the same author and d_u is a document of unknown or questioned authorship. If $\phi(D_k, d_u) = T$, then the author of D_k is also the author of d_u and if $\phi(D_k, d_u) = F$, then the author of D_k is not the same with the author of d_u. In the case of cross-domain verification, D_k and d_u stem from a different text variety or treat a considerably different content (e.g. topics or themes, genres, registers, etc.). For the present task, we considered the simplest (and most challenging) formulation of the verification task, i.e., we only considered cases where $k = 1$, rendering D_k a singleton; thus, only pairs of documents are examined. Given a training set of such problems, i.e., text pairs, the verification systems of the participating teams had to be trained and calibrated to analyze the authorship of the unseen text pairs (from the test set). We shall distinguish between same-author text pairs (SA: $\phi(D_k, d_u) = T$) and different-author (DA: $\phi(D_k, d_u) = F$) text pairs.

A novelty this year is that, contrary to recent editions, we sought to substantially increase the size of the evaluation dataset. This goal was inspired by the observation that previous competitions attracted relatively few approaches that exploited recent advances from representation learning on the basis of neural networks (such as sentence-level embeddings). By supplying large evaluation data, we hoped to broaden up the array of submitted approaches (which seems to have been successful). As every year, we asked participants to deploy the verifiers on TIRA for reproducibility sake and for blind evaluation on an unseen test set [28]. They were expected to produce a score in the form of a bounded scalar between 0 and 1, indicating the probability of the test item being a same-author (SA) pair (rather than a binary choice). The critical threshold for expressing a positive answer was 0.5 (see Section 4).

3 Datasets

In this edition, we have continued working with 'fanfiction' [20, 18], i.e., fictional texts produced by non-professional authors in the tradition of a specific cultural domain (or 'fandom'), such as a famous author or a specific influential work [14]. We operationalize cross-domain verification as cross-fandom verification, where a fandom can be roughly interpreted as a mixture of topic and literary genre. Fanfiction is abundantly available on the internet, as the fastest growing form of online writing [10], and clearly fits our aims to scale up the datasets in the coming years. This year, two training datasets of different magnitudes ("small" and "large") are provided with text pairs crawled from fanfiction.net, a sharing platform for fanfiction that comes with rich, user-contributed metadata. For the construction of these datasets, we built on the fanfiction.net corpus compiled by Bischoff et al. [6] as a basis. Participants were allowed to submit systems calibrated on either dataset (or both). This way, we hoped to be able to establish the added value of increasing training resources in authorship verification. Only English-language texts were included to push the size of the data to the maximum: a multilingual counterpart of this dataset, including resource-scarcer languages, remains a desideratum.

All texts are normalized with regards to punctuation and white space to avoid textual artifacts [4] and have a length of approximately 21,000 characters. To construct the dataset, we bucketed the texts by author and fandom (topic) to ensure a good mix of the two and, despite the very uneven popularity of fandoms and activity of authors, pre-

vent gross overrepresentation of individual fandoms and authors. For the large dataset, 148,000 same-author (SA) and 128,000 different-authors (DA) pairs were drawn from the fanfiction.net crawl. The SA pairs encompass 41,000 authors of which at least four and not more than 400 have written in the same fandom (median: 29). In total, 1,600 fandoms were selected and each single author has written in at least two, but not more than six fandoms (median: 2).

The pairs were assembled by building all possible $\binom{n}{2}$ pairings of author texts (n being the actual number of texts from this author) without allowing two pairs with the same author *and* fandom. If the source texts were longer than 21,000 characters, different random portions were used in each pair. The DA pairs were built from texts of 250,000 authors of which at least two and not more than 800 (twice the number, since each pair consists of two authors now) have written in the same fandom (median: 51). The number of fandoms is the same and largely overlaps with the SA pairs. Each author has written texts in at least one and not more than three fandoms (median: 1).

The small training set is a subset of the large training set with 28,000 SA and 25,000 DA pairs from the same 1,600 fandoms, but with a reduced author number of 6,400 (4–68 per fandom, median: 7) and 48,500 (2–63 per fandom, median: 38), respectively.

The test dataset contains 10,000 SA and 6,900 DA pairs from 400 fandoms and 3,500 and 12,000 authors, respectively, which are guaranteed to exist in the training sets, either in a different author-fandom relation or in the same author-fandom relation, but always with a previously unseen text. This creates a closed-set authorship identification scenario, which will be broken in the next year with unseen fandoms and authors. The number of authors per fandom ranges from 2–400 in the SA pairs (median: 14) and 4–800 in the DA pairs (median: 20). The number of fandoms per author are 2–6 (median: 2) and 1–6 (median: 1).

4 Evaluation Framework

Performance Measures Because of the considerable size of the datasets, we opted for a combination of four performance measures that each focus on different aspects. For each problem (i.e., individual text pair) in the test set, the participating systems submitted a scalar in the [0,1] range, indicating the probability of this being a same-author (SA) pair. For difficult cases, the systems could submit a score of exactly 0.5, which was equivalent to a non-response [25]. The following measures were used to score the submissions:

- **AUC**: the conventional area-under-the-curve score, in a reference implementation [26],
- **c@1**: a variant of the conventional F1 measure, which rewards systems that leave difficult problems unanswered [25],
- **F1**: the well-known performance measure (*not* taking into account non-answers), in a reference implementation [26],
- **F0.5u**: a newly proposed measure that puts more emphasis on deciding same-author cases correctly [5].

The overall score (used to produce the final ranking) is the mean of the scores of all the evaluation measures.

Baselines We applied two baseline systems (calibrated on the small training set only):

1. The first method calculates the cosine similarities between TFIDF-normalized, character tetragram representations of the texts in a pair. The resulting scores are shifted using a grid search on the calibration data ("naive" baseline). This is a so-called first-order verifier [19].
2. Secondly, we applied a text compression method that, given a pair of texts (t_1 and t_2), calculates the cross-entropy of t_2 using the prediction by partial matching model [12] of t_1 and vice-versa. The mean and absolute difference of the two cross-entropies are used by a logistic regression model to estimate a score in $[0, 1]$ (called "compression" baseline below).

5 Survey of Submissions

The authorship verification task received 13 submissions from 10 participating teams. Below we present a brief overview of the submitted approaches:[1]

Boenninghoff et al. [7] (boenninghoff20) proposed an approach that combines neural feature extraction with statistical modeling: a deep metric learning framework with a Siamese network topology was used to measure the similarity between two documents; the produced features were fed into a probabilistic linear discriminant analysis layer that served as a pairwise discriminator to perform Bayes factor scoring in the learned metric space. To take into account topic influences, the authors applied several preprocessing steps, which included (1) replacing all rare tokens/character types by a placeholder, (2) a sliding window to perform tokenization without sentence boundary detection, (3) adding a contextual prefix (fandom label), and (4) dissembling all predefined document pairs and re-sampling new SA and DA pairs in each epoch to increase the heterogeneity of the training data.

Weerasinghe and Greenstadt [32] (weerasinghe20) extracted stylometric features (including function word frequency, vocabulary richness, character and part-of-speech (POS) tag n-grams, POS tag chunks, and noun/verb phrases) and used the absolute difference between the feature vectors as input to a logistic regression model (small dataset) and a neural network-based model with one hidden layer (large dataset). The applied preprocessing steps consist of tokenization, POS tagging, and generating parse trees; the model optimization was done based on the AUC measure.

Halvani et al. [13] (halvani20) used a list of around 1,000 topic-agnostic words and phrases grouped into certain feature categories (e.g., n-grams, sentence starters and endings). Based on such categories, all possible ensembles of feature categories and corresponding thresholds (which were calculated based on the equal error rate of the computed distances for each feature category) were evaluated and the optimal ensemble was selected. The classification was done using the Manhattan metric.

[1] All but two of the submissions (Niven and Kao (niven20) and Faber and van der Ree (faber20)) are described in the participants' notebook papers. Three of the teams (boenninghoff20, araujo20 and weerasinghe20) submitted two systems, calibrated on the small and the large training datasets, respectively; team ordonez20 only used the large dataset; all others used the small one (or even smaller portions of it, as indicated in private correspondence with the organizers).

Kipnis [21] (kipnis20) addressed the task using an unsupervised approach, which takes word-by-word p-values calculated based on a binomial allocation model of words between the two documents, and combines them into a single score using the higher criticism statistic [9]. The produced score was converted into a similarity score by evaluating the empirical distribution of the higher criticism associated with document pairs.

Araujo-Pino et al. [1] (araujo20) used a Siamese neural network approach [8] that receives as input the character n-gram representation (with n varying from 1 to 3) of the document pairs to be compared. The authors experimented with different hyperparameters and trained the model both on the large and the small dataset, using the AUC score as a reference for fine-tuning. The model trained on the small dataset outperformed the model trained on the large one only by a small margin of 0.015 AUC.

Gągała [11] (gagala20) used a data compression method based on the prediction by partial matching model [12], and compression-based cosine as similarity measure between text samples [29]. The method was extended with a context-free grammar character pre-processing: replacing the most frequent character n-grams ($n = 2$) by a special symbol to reduce the length of the texts and to simplify the distribution of the characters. The authors calibrated their method on a subset of the small dataset, reporting that additional data did not improve the performance of the method.

Ordoñez et al. [24] (ordonez20) used a long-sequence transformer, a recent Longformer model [2], to process the long fanfiction documents with additional layers to learn both text and fandom-specific features. The fandom information (provided as metadata) was incorporated by the use of a multi-task loss function that optimizes for both authorship verification and topic correspondence classification losses. The authors additionally compared the performance of the approach with a character-based convolutional neural network ($(CN)^2$) with self-attention layers, reporting that the Longformer system outperformed $(CN)^2$, and both their baselines by a wide margin, achieving a very high overall score of 0.963 on a held-out subset of the large training corpus. On the official test set, however, only a lower overall score of 0.685 was achieved.

Ikae [16] (ikae20) based their approach on text similarity: two documents were considered as an SA pair if the Labbé similarity value [23] between them exceeded a threshold of 0.5. The 500 most frequent words and punctuation marks were used for a document representation according to their relative frequency.

Overall, while the majority of the participants addressed the task as a binary classification problem, they did so with a large variety approaches using both deep learning and machine learning techniques, as well as unsupervised text similarity-based methods. This variability in the submitted methods can be attributed to the increased size of the dataset developed for the current edition.

6 Evaluation Results

The evaluation results can be found in Table 1, where we distinguish between multiple submissions for the same team using a suffix ('-small' or '-large'). A pairwise significance analysis of the F1-scores according to the approximate randomization test [31] is shown in Table 2.

Table 1. Evaluation results for the shared task on authorship verification at PAN 2020 in terms of area under the curve (AUC) of the receiver operating characteristic (ROC), c@1, F0.5u, F1, and an overall score (which determines the order). "Large" ("small") indicates that the large (small) training dataset was used.

Submission	**AUC**	**c@1**	**F0.5u**	**F1**	**Overall**
boenninghoff20-large	**0.969**	**0.928**	**0.907**	**0.936**	**0.935**
weerasinghe20-large	0.953	0.880	0.882	0.891	0.902
boenninghoff20-small	0.940	0.889	0.853	0.906	0.897
weerasinghe20-small	0.939	0.833	0.817	0.860	0.862
halvani20-small	0.878	0.796	0.819	0.807	0.825
kipnis20-small	0.866	0.801	0.815	0.809	0.823
araujo20-small	0.874	0.770	0.762	0.811	0.804
niven20-small	0.795	0.786	0.842	0.778	0.800
gagala20-small	0.786	0.786	0.809	0.800	0.796
araujo20-large	0.859	0.751	0.745	0.800	0.789
baseline (naive)	0.780	0.723	0.716	0.767	0.747
baseline (compression)	0.778	0.719	0.703	0.770	0.742
ordonez20-large	0.696	0.640	0.655	0.748	0.685
ikae20-small	0.840	0.544	0.704	0.598	0.672
faber20-small	0.293	0.331	0.314	0.262	0.300

Table 2. Significance of pairwise differences in F1 between submissions. Notation: '=' (not significantly different: $p > 0.5$), '*' (significantly different: $p < 0.05$), '**' (very significantly different: $p < 0.01$), '***' (highly significantly different: $p < 0.001$).

	boenninghoff20-large	weerasinghe20-large	boenninghoff20-small	weerasinghe20-small	halvani20-small	kipnis20-small	araujo20-small	niven20-small	gagala20-small	araujo20-large	*baseline (naive)*	baseline *(compression)*	ordonez20-large	ikae20-small	faber20-small
boenninghoff20-large		***	***	***	***	***	***	***	***	***	***	***	***	***	***
weerasinghe20-large			***	***	***	***	***	***	***	***	***	***	***	***	***
boenninghoff20-small				***	***	***	***	***	***	***	***	***	***	***	***
weerasinghe20-small					***	***	***	***	***	***	***	***	***	***	***
halvani20-small						=	=	***	=	=	***	***	***	***	***
kipnis20-small							**	***	=	=	***	***	***	***	***
araujo20-small								***	**	***	***	***	***	***	***
niven20-small									***	***	***	***	***	***	***
gagala20-small										=	***	***	***	***	***
araujo20-large											***	***	***	***	***
baseline (naive)												=	=	***	***
baseline (compression)													=	***	***
ordonez20-large														***	***
ikae20-small															***

A number of worthwhile observations can be made. First of all, the top-performing method (boenninghoff20-large) reaches an impressive overall score of 0.935 that significantly outperforms the runner-up, weerasinghe20, which already has a very high overall score of 0.902. The difference between both approaches might correspond to their respective treatment of non-answers, since the difference in performance is the most pronounced for the c@1 and the F1 measures (the latter explicitly ignored the non-answers, which seems to have given boenninghoff20 a competitive edge). Interestingly, both Boenninghoff et al.'s and Weerasinghe and Greenstadt's systems were calibrated on the large dataset and outperform their counterpart trained on the small dataset. While in itself this seems a clear indication that verification systems can benefit from large training datasets, surprisingly enough, this is not in line with the result for araujo20 (where the 'small' system outperformed the 'large' one). Most systems, except three (ordonez20, ikae20 and faber20), outperformed the naive and compression baselines (which furthermore did not produce significantly different results from one another). Interestingly, the cohort following the top-2 performing systems in the final ranking (halvani20-small, kipnis20-small, araujo20-small) achieved overall scores in a highly similar ballpark (in the 0.80s), but often produced only mildly significantly different predictions.

7 Discussion

In this section, we provide a more in-depth analysis of the submitted approaches and their evaluation results. First, we take a look into the distribution of the submitted verification scores, including that of a meta classifier. We go on to inspect the effect of non-responses, and finally, study how the topic similarity between the text in a test pair might have affected the results.

Distributions In Figure 1 (left), we plot the precision-recall curves for all "small" submissions, including that of a meta classifier that predicts the mean score over all submissions (dotted line). The figure highlights the relatively large head start of boeninghoff20 and weerasinghe20 regarding AUC in comparison to the runner-up cohort. The latter achieves a higher precision, where the former capitalizes on a solid recall.

In Figure 1 (right), the distribution of the scores submitted by the teams are visualized by kernel-density estimates for the individual "small" submissions, as well as the overall distribution (dotted line). The latter clearly shows the effect of so-called "number heaping," with modes at predictable thresholds (0, 0.25, 0.5, 0.75 and 1). Furthermore, the modes clearly suggest that the most successful systems have tended towards self-confident scores, close to 0 and 1. Systems with modes in more hesitant areas (e.g., 0.25-0.50) seem to have under-performed in this respect. This might especially have impacted the AUC scores, because this measure favors bold predictions, provided they are correct.

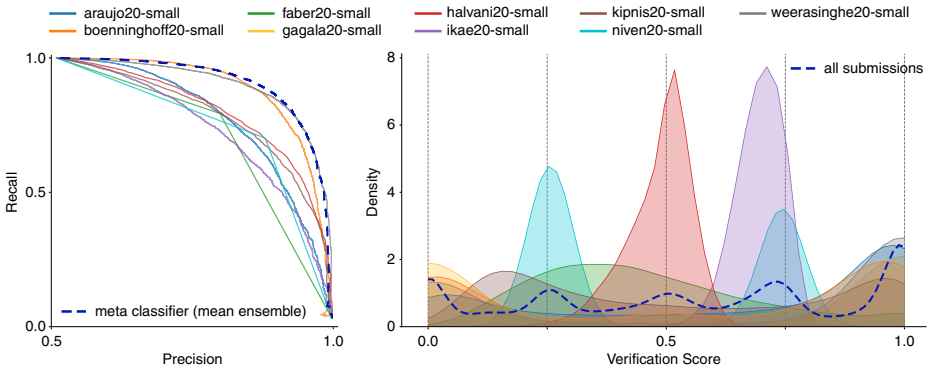

Figure 1. *Left:* Precision-recall curves for all "small" submissions (excluding faber20-small for the sake of readability), as well as a meta classifier that is based on the participants' mean score. *Right:* Kernel-density estimate plots for the distributions of the submitted verification scores (across all and per "small" submission).

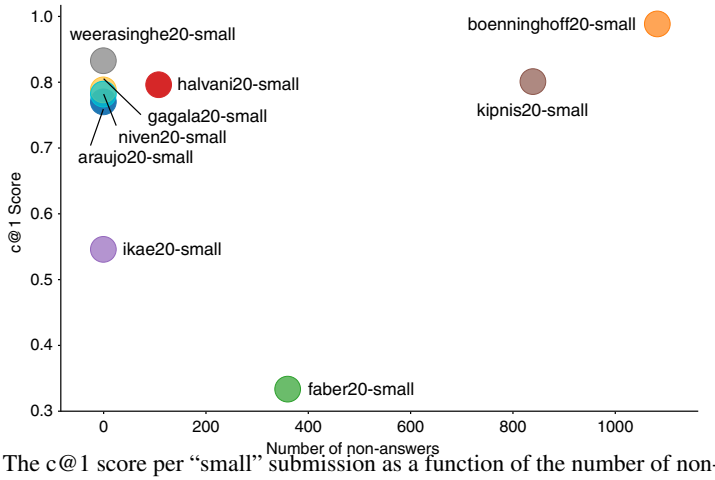

Figure 2. The c@1 score per "small" submission as a function of the number of non-answers.

Non-answers As mentioned above, participants were allowed to leave difficult problems unanswered by giving them a score of exactly 0.5. Such non-responses explicitly affected the c@1 score and were excluded in calculating the F1 score. In Figure 2, we show the correlation between the absolute number of non-answers submitted and the c@1 score. The results show that only three teams made active use of this option: In particular, Boenninghoff et al. and Kipnis submitted a considerable number of non-answers (1,082 and 839, respectively), without compromising their overall score. An interesting follow-up question is then, whether Boenninghoff et al.'s top rank is *solely* due to this non-response strategy. The results in Table 3 suggest otherwise: Here, we recomputed the evaluation measures, excluding those pairs for which boenninghoff20-large submitted a non-response. The previous ranking is corroborated (and even reinforced), assuring us that the difference in performance is not solely due to non-responses.

Table 3. Evaluation results for two top-performing systems, excluding the pairs for which boenninghoff20-large submitted a non-response.

Submission w/o non-responses	AUC	c@1	F1	F0.5u	Overall
boenninghoff20-large	0.974	0.930	0.936	0.934	0.943
weerasinghe20-large	0.957	0.886	0.897	0.888	0.907

Figure 3. Word cloud visualizations of 9 cherry-picked dimensions from the NMF model (60 terms per topic).

The influence of topic We have trained a reference topic model on the small training dataset, in order to be able to trace the influence of topic on the submitted verifiers. Our approach was as follows: The entire corpus was tokenized and POS-tagged using Spacy's standard model for English [15], retaining only nouns, adjectives and verbs. Next, a TF-IDF-normalized bag-of-words representation of the corpus was constructed, considering the 5,000 tokens with the highest cumulative corpus frequency, ignoring words that appeared in more than half of the documents or words with an absolute document frequency of less than 100. Next, we fitted a non-negative matrix factorization (NMF) model of 150 dimensions on this data (during 50 iterations). All modeling was done in scikit-learn [26]. Figure 3 shows word clouds for a cherry-picked subset from the (generally clearly interpretable) topics that were obtained from the model.[2] Finally, this pipeline was applied to the text pairs in the test set and we recorded the cosine similarity between the L1-normalized topic representations for each text. This scalar was used as a proxy for the topic resemblance between two texts.

In Figure 4, the average verification score of the submitted "small" systems for each pair in the test set is plotted as a function of the topic similarity observed for that pair. The fit of a linear model to this data suggests that there is a considerable correlation ($\beta = 0.28, R^2 = 0.16$) between the topic similarity and the likelihood that a system

[2]Based on Andreas Mueller's `word_cloud` package: https://github.com/amueller/word_cloud.

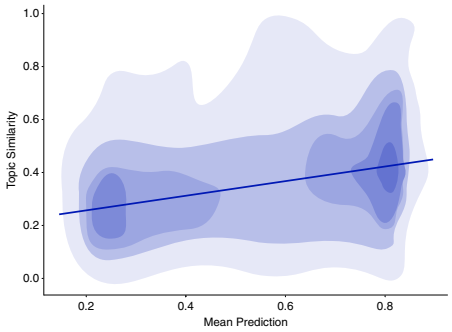

Figure 4. Topical resemblance between a text pair as a function of the mean prediction by all "small" systems ($\beta = 0.28, R^2 = 0.16$).

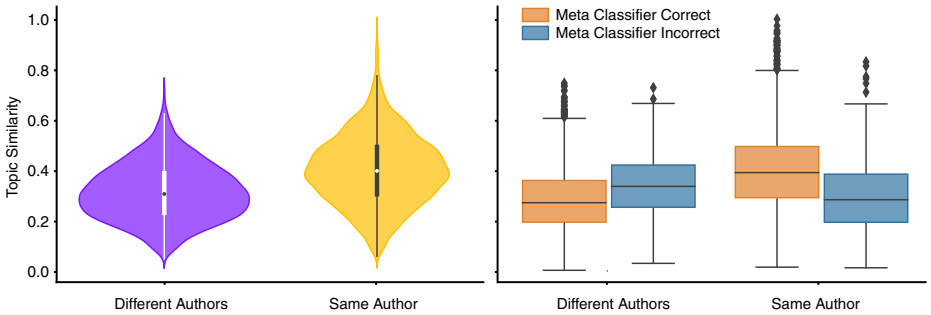

Figure 5. The distribution of topical similarity, separated over SA and DA pairs and whether the meta-classifier correctly solved the pairs.

categorizes the pair as an SA instance. This is not necessarily a bad thing, since, overall, SA pairs show a greater topic similarity than the DA pairs, as shown in the violin plot in Figure 5. Intuitively, this shows that authors tend to write about the same topics.

However, if we split out the data over correct and incorrect cases, the picture changes: In cases where the meta-classifier proposed a correct solution, the numbers are largely unaltered ($\beta = 0.28, R^2 = 0.17$); for the incorrect decisions, however, the explanatory value of the linear model plummets ($\beta = 0.19, R^2 = 0.03$). This suggests that for the incorrect cases, the models were generally susceptible to a misleading influence of topic similarity. This hypothesis is supported by the boxplots in Figure 5. Interestingly, DA pairs which were correctly solved have a lower topical similarity than those that were incorrectly solved (for the SA cases, the relationship is inversed). A nuanced, yet plausible picture emerges from these results: Systems can (and even should) partially rely on topical information when performing large-scale authorship verification, but they should not exaggerate this reliance, as it can be misleading.

8 Perspective

This year's track on authorship identification at PAN focused on authorship verification. The 2020 edition fits an ambitious, renewed three-year strategy that aims to increase the available benchmark resources in both scope and realism. The initiative continues to attract a healthy number of high-quality submissions (this year, we received 13 submissions by 10 teams). The evaluation measures used here attest to the interesting diversity of the submitted systems, including, for the first time, a number of data-hungry approaches based on deep learning, that, surprisingly enough, remained relatively absent from this community for quite some years. The top-performing methods could boast an impressive performance—we recorded an overall score of 0.935 for the highest-ranking submission by Boenninghoff et al. [7]—, showing that the field might be closing in on a solution for authorship verification in a closed setup. Our analyses, however, also suggested that the dependence on topical information is still problematic, and that it is one of the main causes driving misclassifications. This observation considerably ups the stakes for next year's edition, that will present a much more challenging task, involving an open setup, with a test set of unseen authors, as well as unseen domains. In this regard, the recent advances in adversarial training for authorship attribution [6] may prove to be a promising direction for future research in verification as well.

Acknowledgements

We would like to thank all participants for their much-appreciated contribution to the shared task and we wish to encourage them to stay involved in the community in the next years. We thank the CLEF organizers for their work in organizing the conference, especially in these trying times. Our special thanks go to Sebastian Bischoff, Niklas Deckers, Marcel Schliebs, and Ben Thies for sharing the fanfiction.net corpus.

Bibliography

[1] Araujo-Pino, E., Gómez-Adorno, H., Fuentes-Pineda, G.: Siamese network applied to authorship verification. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)

[2] Beltagy, I., Peters, M.E., Cohan, A.: Longformer: The long-document transformer (2020)

[3] Bevendorff, J., Ghanem, B., Giachanou, A., Kestemont, M., Manjavacas, E., Potthast, M., Rangel, F., Rosso, P., Specht, G., Stamatatos, E., Stein, B., Wiegmann, M., Zangerle, E.: Shared tasks on authorship analysis at pan 2020. In: Jose, J.M., Yilmaz, E., Magalhães, J., Castells, P., Ferro, N., Silva, M.J., Martins, F. (eds.) Advances in Information Retrieval, pp. 508–516, Springer International Publishing, Cham (2020), ISBN 978-3-030-45442-5

[4] Bevendorff, J., Stein, B., Hagen, M., Potthast, M.: Bias Analysis and Mitigation in the Evaluation of Authorship Verification. In: Korhonen, A., Màrquez, L., Traum, D. (eds.) 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019), pp. 6301–6306, Association for Computational Linguistics (Jul 2019), URL https://www.aclweb.org/anthology/P19-1634

[5] Bevendorff, J., Stein, B., Hagen, M., Potthast, M.: Generalizing unmasking for short texts. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pp. 654–659, Association for Computational Linguistics (2019), https://doi.org/10.18653/v1/n19-1068, URL https://doi.org/10.18653/v1/n19-1068
[6] Bischoff, S., Deckers, N., Schliebs, M., Thies, B., Hagen, M., Stamatatos, E., Stein, B., Potthast, M.: The importance of suppressing domain style in authorship analysis. CoRR **abs/2005.14714** (2020), URL https://arxiv.org/abs/2005.14714
[7] Boenninghoff, B., Rupp, J., Nickel, R.M., Kolossa, D.: Deep bayes factor scoring for authorship verification. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)
[8] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "Siamese" time delay neural network. In: Cowan, J.D., Tesauro, G., Alspector, J. (eds.) Advances in Neural Information Processing Systems 6, pp. 737–744, Morgan-Kaufmann (1994), URL http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network.pdf
[9] Donoho, D., Jin, J.: Higher criticism for detecting sparse heterogeneous mixtures. The Annals of Statistics **32**(3), 962–994 (2004)
[10] Fathallah, J.: Fanfiction and the Author. How FanFic Changes Popular Cultural Texts. Amsterdam University Press (2017)
[11] Gągała, Ł.: Authorship verification with prediction by partial matching and context-free grammar. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)
[12] Halvani, O., Graner, L.: Cross-domain authorship attribution based on compression: Notebook for PAN at CLEF 2018. In: Cappellato, L., Ferro, N., Nie, J., Soulier, L. (eds.) Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10-14, 2018, CEUR Workshop Proceedings, vol. 2125, CEUR-WS.org (2018), URL http://ceur-ws.org/Vol-2125/paper_90.pdf
[13] Halvani, O., Graner, L., Regev, R.: Cross-domain authorship verification based on topic agnostic features. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)
[14] Hellekson, K., Busse, K. (eds.): The Fan Fiction Studies Reader. University of Iowa Press (2014)
[15] Honnibal, M., Montani, I.: spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing (2017), to appear
[16] Ikae, C.: UniNE at PAN-CLEF 2020: Author verification. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)
[17] Juola, P.: Authorship attribution. Foundations and Trends in Information Retrieval **1**(3), 233–334 (2006)
[18] Kestemont, M., Stamatatos, E., Manjavacas, E., Daelemans, W., Potthast, M., Stein, B.: Overview of the Cross-domain Authorship Attribution Task at PAN 2019. In: Cappellato, L., Ferro, N., Losada, D., Müller, H. (eds.) CLEF 2019 Labs and Workshops, Notebook Papers, CEUR-WS.org (Sep 2019), URL http://ceur-ws.org/Vol-2380/
[19] Kestemont, M., Stover, J.A., Koppel, M., Karsdorp, F., Daelemans, W.: Authenticating the writings of julius caesar. Expert Systems with Applications **63**, 86–96 (2016),

https://doi.org/10.1016/j.eswa.2016.06.029, URL https://doi.org/10.1016/j.eswa.2016.06.029

[20] Kestemont, M., Tschuggnall, M., Stamatatos, E., Daelemans, W., Specht, G., Stein, B., Potthast, M.: Overview of the author identification task at PAN-2018: cross-domain authorship attribution and style change detection. In: Working Notes Papers of the CLEF 2018 Evaluation Labs. Avignon, France, September 10-14, 2018/Cappellato, Linda [edit.]; et al., pp. 1–25 (2018)

[21] Kipnis, A.: Higher criticism as an unsupervised authorship discriminator. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)

[22] Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. Journal of the American Society for Information Science and Technology **60**(1), 9–26 (2009)

[23] Labbé, C., Labbé, D.: Inter-textual distance and authorship attribution Corneille and Molière. Journal of Quantitative Linguistics **8**(3), 213–231 (2001), https://doi.org/10.1076/jqul.8.3.213.4100, URL https://doi.org/10.1076/jqul.8.3.213.4100

[24] Ordoñez, J., Soto, R.R., Chen, B.Y.: Will longformers PAN out for authorship verification? In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)

[25] Peñas, A., Rodrigo, A.: A simple measure to assess non-response. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, p. 1415–1424, HLT '11, Association for Computational Linguistics, USA (2011), ISBN 9781932432879

[26] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)

[27] Potthast, M., Braun, S., Buz, T., Duffhauss, F., Friedrich, F., Gülzow, J.M., Köhler, J., Lötzsch, W., Müller, F., Müller, M.E., Paßmann, R., Reinke, B., Rettenmeier, L., Rometsch, T., Sommer, T., Träger, M., Wilhelm, S., Stein, B., Stamatatos, E., Hagen, M.: Who wrote the web? revisiting influential author identification research applicable to information retrieval. In: Ferro, N., Crestani, F., Moens, M.F., Mothe, J., Silvestri, F., Di Nunzio, G.M., Hauff, C., Silvello, G. (eds.) Advances in Information Retrieval, pp. 393–407, Springer International Publishing, Cham (2016), ISBN 978-3-319-30671-1

[28] Potthast, M., Gollub, T., Wiegmann, M., Stein, B.: TIRA integrated research architecture. In: Ferro, N., Peters, C. (eds.) Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF, The Information Retrieval Series, vol. 41, pp. 123–160, Springer (2019), https://doi.org/10.1007/978-3-030-22948-1_5, URL https://doi.org/10.1007/978-3-030-22948-1_5

[29] Sculley, D., Brodley, C.E.: Compression and machine learning: A new perspective on feature space vectors. In: Proceedings of the Data Compression Conference, p. 332, DCC '06, IEEE Computer Society, USA (2006), ISBN 0769525458, https://doi.org/10.1109/DCC.2006.13, URL https://doi.org/10.1109/DCC.2006.13

[30] Stamatatos, E.: A survey of modern authorship attribution methods. JASIST **60**(3), 538–556 (2009), https://doi.org/10.1002/asi.21001, URL https://doi.org/10.1002/asi.21001

[31] W. Noreen, E.: Computer-Intensive Methods for Testing Hypotheses: An Introduction. A Wiley-Interscience publication (1989)

[32] Weerasinghe, J., Greenstadt, R.: A machine learning model for authorship verification. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, CEUR-WS.org (2020)