VALLA: Standardizing and Benchmarking Authorship Attribution and Verification Through Empirical Evaluation and Comparative Analysis

Jacob Tyo
Army Research Laboratory
Carnegie Mellon University
jacob.p.tyo.civ@army.mil

Bhuwan Dhingra
Duke University
bdhingra@cs.duke.edu

Zachary C. Lipton
Carnegie Mellon University
zlipton@cmu.edu

Abstract

Despite decades of research on authorship attribution (AA) and authorship verification (AV), inconsistent dataset splits/filtering and mismatched evaluation methods make it difficult to assess the state of the art. In this paper, we present a survey of the fields, resolve points of confusion, introduce VALLA that standardizes and benchmarks AA/AV datasets and metrics, provide a large-scale empirical evaluation, and provide apples-to-apples comparisons between existing methods. We evaluate eight promising methods on fifteen datasets (including distribution shifted challenge sets) and introduce a new dataset based on texts archived by Project Gutenberg. Surprisingly, we find that a traditional Ngram-based model performs best on 5 (of 7) AA tasks, achieving an average macro-accuracy of 76.50% (compared to 66.71% for a BERT-based model). However, on the two AA datasets with the greatest number of words per author, as well as on the AV datasets, BERT-based models perform best. While AV methods are easily applied to AA, they are seldom included as baselines in AA papers. We show that through the application of hard-negative mining, AV methods are competitive alternatives to AA methods. VALLA and all experiment code can be found here: https://github.com/JacobTyo/Valla

1 Introduction

The statistical analysis of variations in literary style between one writer or genre and another, commonly known as *stylometry*, dates back as far as 500 AD. Computer-assisted stylometry first emerged in the early 1960s, when Mosteller and Wallace (1963) explored the foundations of computer-assisted authorship analysis. Today automated tools for authorship analysis are common, finding practical use in the justice system to analyze evidence (Koppel et al., 2008), among social media companies to detect compromised accounts (Barbon et al., 2017), to link online accounts belonging one individual (Sinnott and Wang, 2021), and to detect plagiarism (Stamatatos and Koppel, 2011).

In the modern Natural Language Processing (NLP) literature, two problem formulations dominate the study of methods for determining the authorship of anonymous or disputed texts: Authorship Attribution (AA) and Authorship Verification (AV). In AA, the learner is given representative texts for a canonical set of authors in advance, and expected to attribute a new previously unseen text of unknown authorship to one of these a priori known authors. In AV, the learner faces a more general problem: given two texts, predict whether or not they were written by the same author.

While both problems have received considerable attention (Murauer and Specht, 2021; Altakrori et al., 2021; Kestemont et al., 2021), the state of the art is difficult to assess owing to inconsistencies in the datasets, splits, performance metrics, and variations in the framing of domain shift across studies. For example, a recent survey paper (Neal et al., 2017) indicates that the state-of-the-art method is based on the Prediction by Partial Matching (PPM) text compression scheme and the cross-entropy of each text with respect to the PPM categories. By contrast, the PAN-2021 competition (Kestemont et al., 2021) indicates that the state of the art is a hierarchical bi-directional LSTM with learned-CNN text encodings. Recent work (Fabien et al., 2020) concludes that the transformer-based language model BERT is the highest-performing AA method. A recent analysis paper (Altakrori et al., 2021) argue that the traditional approach of character n-grams and masking remains the best methodology to this day. Each of these sources compares methods against different baselines, on different datasets (sometimes on just a single small dataset), and with different problem variations (such as cross-topic, cross-genre, etc.).

In this paper, we start by sorting out this fragmented prior work through a brief survey of the

Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 649–660
November 1–4, 2023.

literature. Then, to present a unified evaluation, we introduce VALLA. VALLA provides standardized versions of all the common AA and AV datasets with uniform evaluation metrics and standardized domain-shifted test sets, and implementations of all methods used in this paper. Additionally, we introduce a new large-scale dataset based on public domain books sourced from Project Gutenberg for both tasks. Then using this benchmark, we present an extensive evaluation of eight common AA and AV methods on their respective datasets with and without domain shift. We also make comparisons *between* AA and AV methods where applicable.

Recent work indicates that traditional methods still outperform pretrained language models (i.e. BERT) (Kestemont et al., 2021; Altakrori et al., 2021; Murauer and Specht, 2021; Tyo et al., 2021; Peng et al., 2021; Futrzynski, 2021), but we show that this narrative only appears to apply to datasets with a limited number of words per class. Furthermore, BERT-based models achieve new state-of-the-art macro-accuracy on the IMDb62 (98.80%) and Blogs50 (74.95%) datasets and set the benchmark on our newly introduced Gutenberg dataset.

The applicability of AV methods to AA problems is frequently mentioned, yet these methods are not placed in competition. We provide this comparison and find that AA methods to outperform AV methods on AA problems, *but only until* hard-negative mining is used during AV training. Initially, AA outperform AV methods by 15% macro-accuracy, but hard negative mining improves the performance of AV models in the AA setting, increasing the macro-accuracy of BERT_V (a verification formulation of the BERT model) to 72.42% on the tested dataset, making it a competitive alternative. In summary, we contribute the following:

- A survey of AA and AV.
- A benchmark that standardizes AA and AV datasets and method implementations
- State-of-the-art accuracy on the IMDb62 (98.80%) and Blogs50 (74.95%) datasets.
- A new dataset with long average text length.
- An evaluation of eight high-performing AA and AV methods on fifteen datasets
- Evidence of the importance of hard-negative mining for authorship applications.

2 Brief Survey of the Literature

Neal et al. (2017) provide an overview of AA dataset characteristics and traditional AA methods. The authors enumerate the wide array of textual features used for AA and provide an evaluation of these techniques on a single, small dataset. They conclude that the prediction using partial matching (PPM) method is the state of the art. Bouanani and Kassou (2014) provide a similar survey focusing on the enumeration of AA hand-engineered features. Stamatatos (2009) discuss traditional AA methods from an instance-based (one text vs another) vs a profile-based (one text vs all authors) methodology, and include a computational requirement analysis.

Among notable surveys, Mekala et al. (2018) compare the benefits of the different traditional textual features; Argamon (2018) detail the problems with applying many traditional AA methods in real-world scenarios; Alhijawi et al. (2018) provide a meta-analysis of the field; and Ma et al. (2020) point out the lack of advances from using transformer-based language models in AA. Critically, all of these prior surveys exclude recent advances due to deep learning, such as recurrent neural networks, transformers, word embeddings, and byte-pair encoding. In this section, we briefly cover more traditional techniques, and then discuss recent deep-learning-based approaches.

So far, we have outlined the work on AA surveys, but there are none to be found that focus on AV. The PAN competition overview (Kestemont et al., 2021) is close, but limited to what appears in competition. Also of note, each year's competition focuses on a single dataset that changes every year.

2.1 Datasets

Murauer and Specht (2021) worked towards a benchmark for AA. They do not discuss AV or the domain shift present in many popular datasets. The test sets often contain novel topics (cross-topic - $\times_t$), genres (cross-genre - $\times_g$), or authors (unique authors - $\times_a$). Table 1 shows the statistical variability between the different datasets. The number of authors, documents, and words in a corpus is influential, but looking more closely at the number of documents per author (D/A) and the number of words per document (W/D) gives a better idea of how hard a corpus is. The larger the number of authors and the less text there is to work with, the harder the problem. Lastly, we measure the imbalance (imb) of datasets based on the standard deviation of the number of documents per author. The CCAT50 (Lewis et al., 2004), CMCC (Goldstein et al., 2008), Guardian (Stamatatos, 2013),

Dataset	Text Type	Typical Setting	iid	$\times_t$	$\times_g$	$\times_a$	D	A	W	D/A	W/D	imb
CCAT50	News	AA	✓	—	—	—	5k	50	2.5M	100	506	0
CMCC	Various	AA	✓	✓	✓	—	756	21	454k	36	601	0
Guardian	Opinion	AA	✓	✓	✓	—	444	13	467k	34	1052	6.7
IMDb62	Reviews	AA	✓	—	—	—	62k	62	21.6M	1000	349	2.6
Blogs50	Blogs	AA	✓	—	—	—	66k	50	8.1M	1324	122	553
BlogsAll	Blogs	AV	✓	—	—	—	520k	14k	121.6M	37	233	90
PAN20 & 21	Various	AV	✓	✓	—	✓	443k	278k	1.7B	1.6	3922	2.3
Amazon	Reviews	AV	✓	✓	—	—	1.46M	146k	91.9M	10	63	0
Gutenberg	Books	AA	✓	—	—	✓	29k	4.5k	1.9B	6	66350	10.5

Table 1: An overview of datasets used for Authorship Attribution (AA) and Authorship Verification (AV). iid is an i.i.d. split, $\times_t$ is a cross-topic split, $\times_g$ is a cross-genre split, $\times_a$ is an unknown author split, D is the number of documents, A is the number of authors, W is the number of words, W/D is the average length of documents, D/A is the average number of documents per author, W/D is the average number of words per document, and imb is the imbalance of the dataset measured by the standard deviation of the number of documents per author. ✓ indicates necessary data is available to create a standardized split, whereas — indicates it isn't.

IMDb62 (Seroussi et al., 2014), and PAN20 & PAN21 (Kestemont et al., 2021) are used as they are in prior work, but with the distinction that we publish our train/validation/test splits to ensure comparability with future work.

Although the Blogs50 dataset (Schler et al., 2006) is common (BlogsALL in Table 1), the statistics we present are different than those originally published. This discrepancy is due to a large number of exact duplicates (∼160,000) which we have removed. The most common form of this dataset is Blogs10 and Blogs50 (the texts only from the "top" 10 and 50 authors respectively). This is problematic because it isn't clear how these "top" authors are selected: the number of documents (Fabien et al., 2020; Patchala and Bhatnagar, 2018), the number of words, with minimum text length (Koppel et al., 2011), with spam (or other) filtering (Yang and Chow, 2014; Halvani et al., 2017), or as in most cases, not specified (Jafariakinabad and Hua, 2022; Yang et al., 2018; Zhang et al., 2018; Ruder et al., 2016). In our framework, we release standard splits and cleaning for this dataset.

Finally, we introduce the Gutenberg authorship dataset, as a new large-sclase authorship corpus with very long texts (each texts is about 17 times longer, on average, than the next longest corpus). While some prior work has used Project Gutenberg[1] as a dataset source (public domain books), they all use small subsets (Arun et al. (2009) use 10 authors, Gerlach and Font-Clos (2020) use the 20 most prolific authors, Menon and Choi (2011) use 14 authors, Rhodes (2015) use 6 authors, Khmelev and Tweedie (2001) get a 380 text subset, etc.). Here we have collected all single-author English texts from Project Gutenberg resulting in almost 2 billion words and a very long average document length.

[1] https://www.gutenberg.org

2.2 Metrics

One of the difficulties in comparing prior work is the use of different performance metrics. Some examples are accuracy (Altakrori et al., 2021; Stamatatos, 2018; Jafariakinabad and Hua, 2022; Fabien et al., 2020; Saedi and Dras, 2021; Zhang et al., 2018; Barlas and Stamatatos, 2020), F1 (Murauer and Specht, 2021), C@1 (Bagnall, 2015), recall (Lagutina, 2021), precision (Lagutina, 2021), macro-accuracy (Bischoff et al., 2020), AUC (Bagnall, 2015; Pratanwanich and Lio, 2014), R@8 (Rivera-Soto et al., 2021), and the unweighted average of F1, F0.5u, C@1, and AUC (Manolache et al., 2021; Kestemont et al., 2021; Tyo et al., 2021; Futrzynski, 2021; Peng et al., 2021; Bönninghoff et al., 2021; Boenninghoff et al., 2020; Embarcadero-Ruiz et al., 2022; Weerasinghe et al., 2021).

In AA and AV, we want to understand the discriminative power of each model, while avoiding metrics that are influenced too much by performance on a small subset of prolific authors. Thus, we adopt *macro-averaged accuracy* for AA (referred to as macro-accuracy), and *AUC* for AV.

2.3 Methods

Figure 1 depicts our categorization.

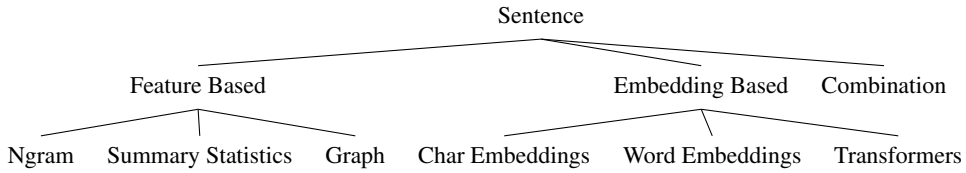

Figure 1: Hierarchy of feature extraction methods

2.3.1 Feature Based

Ngram The most commonly seen input representation (feature) used in AA and AV problems are of N-grams. N-grams provide a fast and simple vectorization method for text that ignores order, based on a given vocabulary of tokens. Granados et al. (2011) introduced *text distortion*, which substitutes out-of-vocabulary items for a "*". Stamatatos (2018) and Bischoff et al. (2020) further test these distortion methods and more complex domain-adversarial methods, showing that the simpler distortion methods are most effective.

The Ngram-based *unmasking* method (Koppel and Schler, 2004), is based on the idea that the style of texts from the same author differs only in a few features. At its core, this method iteratively trains classifiers to predict if two texts are from the same author, but with a decreasing number of features at each round. Then based on the accuracy degradation, a prediction of the same or different author is made. Similarly, Koppel et al. (2011) keep score of how often each author is predicted after random subsets of features are selected, and then make a final prediction based on these scores, dubbed the imposter's method, and Bevendorff et al. (2019) use oversampling with this method to deal with short texts.

Seroussi et al. (2011) use Latent Dirichlet Allocation (LDA), comparing the distance between text representations to determine authorship. They find that this topic-modeling approach can be competitive with the imposter's method while requiring less computation. Seroussi et al. (2014) expand on this topic model approach, and while they present good results on the PAN'11 dataset, the performance of the topic modeling approaches lags behind the best methods. Zhang et al. (2018) introduce a high-performing method that leverages sentence syntax trees and character n-grams as input to a CNN. Saedi and Dras (2021) also presents good results with CNN models, but Ordoñez et al. (2020) indicate that these CNN methods are no longer competitive.

Summary Statistics While older methods focused on small sets of summary statistics, more modern methods are able to combine all of these into a single model. Weerasinghe et al. (2021) provide the best example of this, calculating a plethora of handcrafted features and Ngrams for each document (distribution of word lengths, hapax-legomena, Maas' a^2, Herdan's V_m, and more). The authors take the difference between these large feature vectors for two texts and then train a logistic regression classifier to predict if the texts were written by the same author or not. Despite its simplicity, this method performs well.

Co-occurance Graphs Arun et al. (2009) construct a graph that represents a text based on the stopwords (nodes) and the distance between them (edge weights). Then to compare the two texts, their graphs are compared using the Kullback-Leibler (KL) divergence. Embarcadero-Ruiz et al. (2022) also construct a graph for each text but instead represent each node as a [word, POS_tag] tuple, and each vertex indicates adjacency frequency. After the graph is created for each text, it is encoded into a one-hot representation and used as input to a LEConv layer. After pooling, the absolute difference between the two document representations is passed through a fully connected network for final scoring.

2.3.2 Embedding Based

Char Embedding Bagnall (2015) use a character-level recurrent neural network (RNN) for authorship verification by sharing the RNN model across all authors but training a different head for each author in the dataset. To classify authors, they calculate the probability that each text was written by each author, predicting the author with the highest probability. Ruder et al. (2016) use both CNNs to embed characters and words for AA. Their results show that the character-based method outperforms the word-based approach across several datasets. Compression-based methods, which leverage a compression algorithm (such as ZIP, RAR, etc.) to build text representations which are then

compared with a distance metric, fall into this category as well (Halvani et al., 2017).

Word Embedding Bönninghoff et al. (2019) leverage the Fasttext pre-trained word embeddings, concatenated with a learned CNN character embedding, as part of the input to a bi-directional Long Short Term Memory (BiLSTM) network. That output is then used as input to another network to produce a final document embedding. This neural network structure runs in parallel for two documents (i.e. as a Siamese network (Koch et al., 2015)), and then optimized according to the contrastive loss function. This method was introduced by Bönninghoff et al. (2019), and then later modified to include Bayes factor scoring on the output by Boenninghoff et al. (2020), and by Bönninghoff et al. (2021) to include an uncertainty adaptation layer for defining non-responses. This was the highest performing method at the PAN20 and PAN21 competitions (Kestemont et al., 2021).

Jafariakinabad and Hua (2022) build the equivalent of pre-trained word embeddings but for sentence structure (i.e. GloVe-like embeddings that map sentences with a similar structure close together but are agnostic of their meaning), by using the CoreNLP parse-tree and a traditional word-embedded sentence as input to two identical but separate BiLSTMs, and optimize via contrastive loss. The authors also compare against prior work (Jafariakinabad and Hua, 2019) which embeds the POS-tags along with the word embeddings instead of using their custom structural embedding network, showing slight improvement and improved efficiency. CNN's have also been explored given word embeddings as input (Hitschler et al., 2018; Shrestha et al., 2017; Ruder et al., 2016), yet their results are not among the highest.

Transformers Rivera-Soto et al. (2021) build universal representations for AA and AV by exploring the zero-shot transferability of different methods between three different datasets. The authors train a Siamese BERT model (Reimers and Gurevych, 2019) on one dataset and then test the performance on another without updating. Unfortunately, the results seem to indicate more about the underlying datasets then the ability of these models to uncover a universal authorship representation. Manolache et al. (2021) also explore the applicability of BERT to AA by using BERT embeddings as the feature set for the unmasking method. Comparing this to Siamese BERT, Character BERT (El Boukkouri et al., 2020), and BERT for classification, they find that simple fine-tuning outperforms the more complicated unmasking setup.

Following Bagnall (2015), Barlas and Stamatatos (2020) approach the AA problem by using a shared language model with a different network head for each author. They then compare different shared language model architectures (RNN, BERT, GPT2, ULMFiT, and ELMo), finding that pretrained language models improve the performance of the original RNN architecture. However, the results are all from the small CMCC corpus. Tyo et al. (2021) use a Siamese BERT setup with triplet loss and hard-negative mining for training. Futrzynski (2021) concatenate 28 tokens from each text and then use BERT's `[CLS]` output token for author classification. Peng et al. (2021) concatenate 256 tokens from each text to produce a 512 token input for BERT, and then after pooling use linear layers for same/different author prediction. They repeat this 30 times, sampling different sections of the input texts, and then average over the 30 predictions for final classification.

2.3.3 Feature and Embedding Based

Fabien et al. (2020) explore the applicability of BERT to authorship attribution. They combine the output of BERT with summary statistics via a logistic regression classifier, but find that the summary statistics did not boost performance.

3 The VALLA Benchmark

In 1440, Lorenzo Valla proved that the *Donation of Constantine* (where Constantine I gave the whole of the Western Roman Empire to the Roman Catholic Church) was a forgery, using word choice and other vernacular stylistic choices as evidence (Valla, 1922). Inspired by this influential use of AA, we introduce VALLA: A standardized benchmark for authorship attribution and verification.[2] VALLA includes all datasets in Table 1, along with others from prior literature (Klimt and Yang, 2004; Manolache et al., 2022; Overdorf and Greenstadt, 2016; Altakrori et al., 2021), with standardized splits, cross-topic/cross-genre/unique author test sets, and usable in either AA or AV formulation. VALLA also includes five method implementations, and we use the subscript "A" or "V" to distinguish between the attribution and verification model formulations respectively.

[2]Valla can be found here: https://github.com/JacobTyo/Valla

	CCAT50	CMCC	Guardian	IMDb62	Blogs50	PAN20	GutenburgAA	Average
Ngram_A	**76.68**	**86.51**	**100**	**98.81**	72.28	**43.52**	57.69	**76.50**
PPM_A	69.36	62.30	86.28	95.90	72.16	—	—	55.14
BERT_A	65.72	60.32	84.23	**98.80**	**74.95**	23.83	**59.11**	66.71
pALM_A	63.36	54.76	66.67	—	—	—	—	26.40

Table 2: Macro-accuracy (%) of the authorship attribution models. The "Average"column represents the average macro-accuracy of each model across all datasets in this table, where — entries are counted as 0%.

Ngram Being the best performing method in Altakrori et al. (2021), Murauer and Specht (2021), Bischoff et al. (2020), and Stamatatos (2018), this method creates character Ngram, part-of-speech Ngram, and summary statistics for use as input to an ensemble of logistic regression classifiers. For use in the AV setting, we follow Weerasinghe et al. (2021) by using the difference between the Ngram feature vectors of two texts as input to the logistic regression classifier.

PPM Originally developed in Teahan and Harper (2003) and best performing in Neal et al. (2017), this method uses the prediction by partial matching (PPM) compression model (a variant of PPM is used in the RAR compression software) to compute a character-based language model for each author (Halvani and Graner, 2018), and then the cross-entropy between a test text and each author model is calculated. For use in an AV setup, one text is used to create a model and then the cross-entropy is calculated on the second text.

BERT With the highest reported performance on the AA dataset Blogs50 (Fabien et al., 2020) and the most parameters (over 110 million), this method combines a BERT pre-trained language model with a dense layer for classification. For evaluation, we chunk the evaluation text into non-overlapping sets of 512 tokens and take the majority vote of the predictions. For use in the AV setup, the BERT model is used as the base for a Siamese network and trained with contrastive loss (Tyo et al., 2021). For evaluation in the AV setup, we chunk two texts into K sets of 512 stratified tokens (such that the first 512 tokens of each text are compared, the second grouping is compared, etc.), and then take the majority vote of the K predictions.

pALM The best-performing model in Barlas and Stamatatos (2020) was another variation on BERT where a different head was learned on top of the BERT language model for each known author. We refer to this method as the per-Author Language Model (pALM). To classify a text, it is passed through the model for each author, and then the author model with the lowest perplexity on the text is predicted. This is only used in AA formulations as in AV we would have only a single text to train a network head with.

HLSTM Originally introduced by Bönninghoff et al. (2019), this method leverages a hierarchical BiLSTM setup with Fasttext word embeddings and a custom word embedding learned using a character level CNN, as input to a Siamese network. This was the winning method at PAN20 and PAN21 (Kestemont et al., 2021) and is only used in AV formulations. While this can be modified to work in AA, we follow prior work and use it only for AV.

All of these methods fall into two categories: those that predict an author class, and those that predict text similarity. The methods that predict an author class (whether via logistic regression, dense layer, etc.) need no post-processing. However, methods that predict similarity need post-processing both for AA and AV problems. For AA, we build an *author profile* by randomly selecting 10 texts from each author and averaging their embeddings together. Then we can compare the unknown texts to each author profile and predict the author that is most similar (in euclidean space). For AV, we directly compare the text representations (again using euclidean distance) and then define a hard threshold based on a grid search on the evaluation set (although for computing AUC this threshold is irrelevant).

4 Experiments and Discussion

All experiments were carried out on 8 V100 GPUs and consumed over 5,000 GPU hours. We optimized for hyperparameters on the validation set via random search, and report all values in the VALLA codebase. All results reported are from a single run that uses the best hyperparameters and is trained until there was no improvement for 2 epochs.

4.1 The State-of-the-Art in Authorship Attribution

We start by determining model performance on authorship attribution: given data that is directly attributable to a specific author, learn to classify the work of each author well (macro-accuracy). After evaluating all methods in VALLA on the AA datasets listed in Table 1, we find that the traditional Ngram method is the highest performing on average as detailed in Table 2. However, we do see that the BERT_A model closes the gap on (and can even exceed) the performance of the Ngram_A method as the size of the training set increases. This correlation does not hold on the PAN20 dataset, where the best performing model is still Ngram_A. This indicates that the state-of-the-art AA method is dependent upon the number of words per author available. While we do not provide a detailed analysis of the data requirements of each method, our results roughly indicate that Ngram_A is the method of choice for datasets with less than $50,000$ words per author, while BERT_A is the state-of-the-art method for datasets with over $100,000$ words per author. PPM_A is simple to tune due to few hyperparameters, but it is both a low performer and it scales poorly to large datasets (rendering it unusable on the PAN20 and Gutenberg datasets). pALM_A is the lowest performing method tested, is expensive to train, and scales poorly, so we did not get results on the larger datasets.

The macro-accuracy of BERT_A on the IMDb62 and Blogs50 datasets presents a new state of the art, while defining the initial performance marks on the GutenbergAA and PAN20 datasets.[3] The performance on the Blogs50 dataset requires a bit more analysis due to our filtering of duplicates in the dataset. As a better comparison to prior reported performance, we first explore the performance of BERT_A on the Blogs50 dataset *without* the filtering, and achieve a macro-accuracy of 64.3%. This represents the state-of-the-art accuracy on a version of the dataset more comparable with prior work (despite its issues) but indicates the strength of the result reported in Table 2.

Our results on the Guardian and CMCC datasets are hard to compare to prior work due to the previously mentioned standardization issues, most notably a i.i.d. split has not been used in prior work.

[3]These are initial results because the PAN20 competition was formulated as an AV problem, whereas here we use the AA formulation

	CMCC $\times_t$	CMCC $\times_g$	Guard $\times_t$	Guard $\times_g$
Ngram_A	**82.54**	**84.13**	**86.92**	**87.22**
PPM_A	52.38	57.14	69.23	72.08
BERT_A	49.21	45.24	75.64	75.56
pALM_A	57.14	46.03	61.79	47.22

Table 3: Macro-accuracy (%) of the authorship attribution models on domain shifted AA tests sets. $\times_t$ represents cross-topic and $\times_g$ represents cross-genre.

	PAN21	AmaAV	BlogAV	GutAV
Ngram_V	**0.9719**	0.7742	0.5410	0.8741
PPM_V	0.7917	0.6492	0.6230	0.8508
BERT_V	0.9709	**0.8943**	**0.9201**	**0.9624**
HLSTM_V	0.9693	0.8734	0.8580	0.9147

Table 4: AUC of the AV models on the selected AV datasets.

The CCAT50 dataset, on the other hand, is directly comparable to prior work. Currently, we show best performing model as the Ngram. However, Jafariakinabad and Hua (2022) report the accuracy of a CNN that takes the syntactic tree of a sentence as input as 83.2% which is better than what we were able to achieve.[4]

4.2 The State-of-the-Art in Authorship Attribution under Domain Shift

While dealing with domain shift is an open problem, exploration of domain shift in AA and AV settings is common, even if not explicitly recognized. Table 3 examines the performance of the same AA models but focuses on the cross-topic and cross-genre test sets of the CMCC and Guardian datasets. In other terms, the topic (cross-topic) or genre (cross-genre) of the training and test sets are different, therefore giving a lens into how general the models can under such iid violations. Just as in the i.i.d. setting, the Ngram_A method dominates in all scenarios. It should be noted that all datasets used in this domain shift scenario are small, so we cannot verify that the BERT_A method would begin to dominate as the number of words per author increases. We leave the exploration of domain shift performance on larger datasets to future work, although we expect that the BERT_A model would begin to outperform Ngram_A.

	C50	CM	Guard	I62	B50
HLSTM_V	4.56	8.33	27.59	37.82	57.49
(P)HLSTM_V	13.36	16.27	38.97	59.47	11.34
BERT_V	48.64	35.75	27.82	76.62	60.72
(P)BERT_V	56.80	40.87	61.41	73.17	67.21
BERT_A	**65.72**	**60.32**	**84.23**	**98.80**	**74.95**

Table 5: Macro-accuracy (%) of the AV models on AA datasets. The (P) indicates that the model was pretrained on the PAN20 training set before fine-tuned on the corresponding dataset. Here we use the following abbreviations: C50 (CCAT50), CM (CMCC), Guard (Guardian), I62 (IMDb62), B50 (Blogs50).

	CMCC $\times_t$	CMCC $\times_g$	Guard $\times_t$	Guard $\times_g$
HLSTM_V	7.94	3.18	19.23	23.33
(P)HLSTM_V	9.52	5.56	40.00	31.53
BERT_V	28.85	13.49	42.31	46.53
(P)BERT_V	33.33	19.05	43.33	54.72
BERT_A	**49.21**	**45.24**	**75.64**	**75.56**

Table 6: Macro-accuracy (%) of the authorship verification models on the domain shift AA datasets, where $\times_t$ represents cross-topic and $\times_g$ represents cross-genre. The (P) indicates that the model was pretrained on the PAN20 training set before fine-tuned on the corresponding dataset.

Metric (Formulation)	AUC (AV)	Acc (AV)	Mac-Acc (AA)
BERT_V	0.9229	82.33	67.21
BERT_V w/HNM	**0.9276**	**82.72**	**72.42**

Table 7: This table compares the performance of the same model (BERT_V), on the same data (Blogs50), just formulated in different ways, using different performance metrics (column header). w/HNM represents training with hard negative mining.

4.3 The State-of-the-Art in Authorship Verification

Now we determine model performance on authorship verification: given two texts, determine if they were written by the same author or not. Keeping in line with prior work, the distinction between domain shifted datasets is less clear when formulated as an AV problem. The PAN21 test set is comprised of authors that do not appear in the training set. However the remainder of the datasets (AmazonAV, BlogsAV, and GutenbergAV) are all iid dataset splits. Table 4 details the performance of the AV methods on selected AV datasets.

While we saw the Ngram_A method dominating on most AA datasets, here we see that the deep learning-based HLSTM_V and BERT_V methods attain the highest AUC across the board. However, in AV there are only two classes (same and different author), and therefore all of the datasets have a very large number of words per class (vs classes with limited data in AA). Seemingly because of this key difference, AV formulations tend to be more effective for training deep learning methods.

4.4 Comparing AA and AV methods

Despite the prominence of comments indicating how AV is the fundamental problem of AA, there is no evidence of how well their performance actually transfers. Table 5 shows the performance of LSTM_V and BERT_V on the i.i.d. AA datasets, both when trained only on the dataset as well as starting from a pretrained version of the models (the PAN20 training set was used for pretraining). Here, and in Table 6 for the $\times_t$ and $\times_g$ settings, we see notably lower performance than what was obtained by the AA methods. [5]

Hard-Negative Mining We find that AV methods do not necessarily perform well under an AA formulation. To correctly classify a text in the AA setting, a model must make harder comparisons (i.e., compare one text to all others, therefore it will encounter the hardest comparison), whereas an AV setting is strictly easier as it must compare to only a single text. This interpretation motivates the exploration of using hard-negative mining (updating a model during training only on the hardest examples in each batch) for improving the transferability of AV methods to AA problems.

In this section we take a single model (BERT_V) and train two versions of it: one with the contrastive loss and one using triplet loss with batch hard negative mining (specifically the per-batch hard negative mining methodology used in Hermans et al. (2017)). Table 7 details these results, showing two key findings. The first is that high AV AUC does not indicate high AA macro-accuracy, and the second is that training an AV method with hard negative mining has little effect on its AV AUC but drastically improves its AA macro-accuracy.

[4] CCAT50 is a balanced dataset, so the macro-accuracy and accuracy are equal.

[5] We note that the lower performance of the pretrained H-LSTM on Blogs50 than its non-pretrained version is due to the vocabulary selection. This method chooses its vocabulary based on the pretraining corpus, causing transfer issues.

5 Conclusion

After a survey of the AA and AV landscapes, we present VALLA: an open-source dataset and metric standardization benchmark, complete with implementations of all methods used herein. Using VALLA, we present an extensive evaluation of AA and AV methods in a wide variety of common formulations. We achieve a new state-of-the-art macro-accuracy on the IMDb62 (98.81%) and Blogs50 (74.95%) datasets and provide benchmark results on the other datasets.

Our results show that the AV problem formulation is more effective for training deep models. After showing that the high-performing BERT_V does not perform competitively in AA problems, we explore the effect of hard-negative mining on its performance and find that with no degradation in AV performance, it improves the AA macro-accuracy of BERT_V by over 5%, making it a competitive method in the AA formulation. We hope that VALLA makes future work in AA and AV more easily approached, and more easily comparable.

6 Risks and Limitations

The main risks associated with the development and refinement of AA and AV methods is their misuse. The power to accurately attribute a piece of text to its author holds profound implications, both positive and negative, that warrant careful consideration.

From a privacy perspective, an individual's right to anonymity could be compromised by the misuse of AA and AV methods. While in some circumstances the uncovering of an author's identity is beneficial, such as in forensics or in verifying the authenticity of historical documents, the same technology could also be exploited to unmask authors who wish to remain anonymous for personal, political, or safety reasons.

In this work, we evaluate only on the English language. Furthermore, substantial computational resources were used (over 5,000 GPU hours on V100s). Despite this large amount of compute, after extensive hyperparameter searching, we were only able to get a single run to report metrics on and leave understanding more about the distribution of these results to future work. Both a qualitative analysis, and evaluation of the latest release of large language models are also left to future work.

References

Bushra Alhijawi, Safaa Hriez, and Arafat Awajan. 2018. Text-based authorship identification-a survey. In *2018 Fifth International Symposium on Innovation in Information and Communication Technology (ISI-ICT)*, pages 1–7. IEEE.

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

Shlomo Argamon. 2018. Computational forensic authorship analysis: Promises and pitfalls. *Language and Law/Linguagem e Direito*, 5(2):7–37.

Rajkumar Arun, Venkatasubramaniyan Suresh, and CE Veni Madhavan. 2009. Stopword graphs and authorship attribution in text corpora. In *2009 IEEE international conference on semantic computing*, pages 192–196. IEEE.

Douglas Bagnall. 2015. Author identification using multi-headed recurrent neural networks. In *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015*, volume 1391 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Sylvio Barbon, Rodrigo Augusto Igawa, and Bruno Bogaz Zarpelão. 2017. Authorship verification applied to detection of compromised accounts on online social networks. *Multimedia Tools and Applications*, 76(3):3213–3233.

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. Springer.

Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. 2019. Generalizing unmasking for short texts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 654–659, Minneapolis, Minnesota. Association for Computational Linguistics.

Sebastian Bischoff, Niklas Deckers, Marcel Schliebs, Ben Thies, Matthias Hagen, Efstathios Stamatatos, Benno Stein, and Martin Potthast. 2020. The importance of suppressing domain style in authorship analysis. *arXiv preprint arXiv:2005.14714*.

Benedikt T. Boenninghoff, Julian Rupp, Robert M. Nickel, and Dorothea Kolossa. 2020. Deep bayes factor scoring for authorship verification. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020*, volume 2696 of *CEUR Workshop Proceedings*. CEUR-WS.org.

B. Bönninghoff, S. Hessler, D. Kolossa, and R. M. Nickel. 2019. Explainable authorship verification in social media via attention-based similarity learning. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 36–45, Los Alamitos, CA, USA. IEEE Computer Society.

Benedikt T. Bönninghoff, Robert M. Nickel, and Dorothea Kolossa. 2021. O2D2: out-of-distribution detector to capture undecidable trials in authorship verification. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021*, volume 2936 of *CEUR Workshop Proceedings*, pages 1846–1857. CEUR-WS.org.

Sara El Manar El Bouanani and Ismail Kassou. 2014. Authorship analysis studies: A survey. *International Journal of Computer Applications*, 86(12).

Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Jun'ichi Tsujii. 2020. CharacterBERT: Reconciling ELMo and BERT for word-level open-vocabulary representations from characters. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6903–6915, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Daniel Embarcadero-Ruiz, Helena Gómez-Adorno, Alberto Embarcadero-Ruiz, and Gerardo Sierra. 2022. Graph-based siamese network for authorship verification. *Mathematics*, 10(2).

Maël Fabien, Esau 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, Indian Institute of Technology Patna, Patna, India. NLP Association of India (NLPAI).

Romain Futrzynski. 2021. Author classification as pretraining for pairwise authorship verification. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021*, volume 2936 of *CEUR Workshop Proceedings*, pages 1945–1952. CEUR-WS.org.

Martin Gerlach and Francesc Font-Clos. 2020. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *Entropy*, 22(1):126.

Jade Goldstein, Kerri Goodwin, Roberta Sabin, and Ransom Winder. 2008. Creating and using a correlated corpus to glean communicative commonalities. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Ana Granados, Manuel Cebrian, David Camacho, and Francisco de Borja Rodriguez. 2011. Reducing the loss of information through annealing text distortion. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1090–1102.

Oren Halvani and Lukas Graner. 2018. Cross-domain authorship attribution based on compression. *Working Notes of CLEF*.

Oren Halvani, Christian Winter, and Lukas Graner. 2017. On the usefulness of compression models for authorship verification. In *Proceedings of the 12th international conference on availability, reliability and security*, pages 1–10.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*.

Julian Hitschler, Esther Van Den Berg, and Ines Rehbein. 2018. Authorship attribution with convolutional neural networks and pos-eliding. In *Proceedings of the Workshop on Stylistic Variation (EMNLP 2017). September 8, 2017 Copenhagen, Denmark*, pages 53–28. The Association for Computational Linguistics.

Fereshteh Jafariakinabad and Kien A. Hua. 2019. Style-aware neural model with application in authorship attribution. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 325–328.

Fereshteh Jafariakinabad and Kien A. Hua. 2022. A self-supervised representation learning of sentence structure for authorship attribution. *ACM Trans. Knowl. Discov. Data*, 16(4).

Mike Kestemont, Enrique Manjavacas, Ilia Markov, Janek Bevendorff, Matti Wiegmann, Efstathios Stamatatos, Benno Stein, and Martin Potthast. 2021. Overview of the cross-domain authorship verification task at pan 2021. In *CLEF (Working Notes)*, pages 1743–1759.

Dmitri V Khmelev and Fiona J Tweedie. 2001. Using markov chains for identification of writer. *Literary and linguistic computing*, 16(3):299–307.

Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In *European conference on machine learning*, pages 217–226. Springer.

Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, page 0. Lille.

Moshe Koppel and Jonathan Schler. 2004. Authorship verification as a one-class classification problem. In *Proceedings of the twenty-first international conference on Machine learning*, page 62.

Moshe Koppel, Jonathan Schler, and Shlomo Argamon. 2011. Authorship attribution in the wild. *Language Resources and Evaluation*, 45(1):83–94.

Moshe Koppel, Jonathan Schler, and Eran Messeri. 2008. Authorship attribution in law enforcement scenarios. *NATO Security Through Science Series D-Information and Communication Security*, 15:111.

Kseniya Vladimirovna Lagutina. 2021. Comparison of style features for the authorship verification of literary texts. *Modeling and analysis of information systems*, 28(3):250–259.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

Weicheng Ma, Ruibo Liu, Lili Wang, and Soroush Vosoughi. 2020. Towards improved model design for authorship identification: A survey on writing style understanding. *arXiv preprint arXiv:2009.14445*.

Andrei Manolache, Florin Brad, Antonio Barbalau, Radu Tudor Ionescu, and Marius Popescu. 2022. Veridark: A large-scale benchmark for authorship verification on the dark web. *arXiv preprint arXiv:2207.03477*.

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

Sreenivas Mekala, Vishnu Vardan Bulusu, and Raghunadha Reddy. 2018. A survey on authorship attribution approaches. *Int. J. Comput. Eng. Res.(IJCER)*, 8(8).

Rohith Menon and Yejin Choi. 2011. Domain independent authorship attribution without domain adaptation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 309–315.

Frederick Mosteller and David L. Wallace. 1963. Inference in an authorship problem. *Journal of the American Statistical Association*, 58(302):275–309.

Benjamin Murauer and Günther Specht. 2021. Developing a benchmark for reducing data bias in authorship attribution. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 179–188, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tempestt Neal, Kalaivani Sundararajan, Aneez Fatima, Yiming Yan, Yingfei Xiang, and Damon Woodard. 2017. Surveying stylometry techniques and applications. *ACM Comput. Surv.*, 50(6).

Juanita Ordoñez, Rafael Rivera Soto, and Barry Y Chen. 2020. Will longformers pan out for authorship verification. *Working Notes of CLEF*.

Rebekah Overdorf and Rachel Greenstadt. 2016. Blogs, twitter feeds, and reddit comments: Cross-domain authorship attribution. *Proc. Priv. Enhancing Technol.*, 2016(3):155–171.

Jagadeesh Patchala and Raj Bhatnagar. 2018. Authorship attribution by consensus among multiple features. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2766–2777.

Zeyang Peng, Leilei Kong, Zhijie Zhang, Zhongyuan Han, and Xu Sun. 2021. Encoding text information by pre-trained model for authorship verification. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021*, volume 2936 of *CEUR Workshop Proceedings*, pages 2103–2107. CEUR-WS.org.

Naruemon Pratanwanich and Pietro Lio. 2014. Who wrote this? textual modeling with authorship attribution in big data. In *2014 IEEE International Conference on Data Mining Workshop*, pages 645–652. IEEE.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Dylan Rhodes. 2015. Author attribution with cnns. *Avaiable online: https://www.semanticscholar.org/paper/Author-Attribution-with-Cnn-s-Rhodes/ (accessed on 22 August 2016)*.

Rafael A. Rivera-Soto, Olivia Elizabeth Miano, Juanita Ordonez, Barry Y. Chen, Aleem Khan, Marcus Bishop, and Nicholas Andrews. 2021. Learning universal authorship representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 913–919, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. Character-level and multi-channel convolutional neural networks for large-scale authorship attribution. *arXiv preprint arXiv:1609.06686*.

Chakaveh Saedi and Mark Dras. 2021. Siamese networks for large-scale author identification. *Computer Speech & Language*, 70:101241.

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI spring symposium: Computational approaches to analyzing weblogs*, volume 6, pages 199–205.

Yanir Seroussi, Ingrid Zukerman, and Fabian Bohnert. 2011. Authorship attribution with latent dirichlet allocation. In *Proceedings of the fifteenth conference on computational natural language learning*, pages 181–189.

Yanir Seroussi, Ingrid Zukerman, and Fabian Bohnert. 2014. Authorship attribution with topic models. *Computational Linguistics*, 40(2):269–310.

Prasha Shrestha, Sebastian Sierra, Fabio A González, Manuel Montes-y Gómez, Paolo Rosso, and Thamar Solorio. 2017. Convolutional neural networks for authorship attribution of short texts. In *EACL (2)*, pages 669–674.

Richard Sinnott and Zijian Wang. 2021. Linking user accounts across social media platforms. In *2021 IEEE/ACM 8th International Conference on Big Data Computing, Applications and Technologies (BDCAT'21)*, pages 18–27.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556.

Efstathios Stamatatos. 2013. On the robustness of authorship attribution based on character n-gram features. *Journal of Law and Policy*, 21(2):421–439.

Efstathios Stamatatos. 2018. Masking topic-related information to enhance authorship attribution. *Journal of the Association for Information Science and Technology*, 69(3):461–473.

Efstathios Stamatatos and Moshe Koppel. 2011. Plagiarism and authorship analysis: introduction to the special issue. *Language Resources and Evaluation*, 45(1):1–4.

William J. Teahan and David J. Harper. 2003. *Using Compression-Based Language Models for Text Categorization*, pages 141–165. Springer Netherlands, Dordrecht.

Jacob Tyo, Bhuwan Dhingra, and Zachary Lipton. 2021. Siamese bert for authorship verification. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021*, volume 2936 of *CEUR Workshop Proceedings*, pages 2169–2177. CEUR-WS.org.

Lorenzo Valla. 1922. Discourse on the forgery of the alleged donation of constantine. *Latin and English. Trans. Christopher B. Coleman. New Haven*.

Janith Weerasinghe, Rhia Singh, and Rachel Greenstadt. 2021. Feature vector difference based authorship verification for open-world settings. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021*, volume 2936 of *CEUR Workshop Proceedings*, pages 2201–2207. CEUR-WS.org.

Min Yang, Xiaojun Chen, Wenting Tu, Ziyu Lu, Jia Zhu, and Qiang Qu. 2018. A topic drift model for authorship attribution. *Neurocomputing*, 273:133–140.

Min Yang and Kam-Pui Chow. 2014. Authorship attribution for forensic investigation with thousands of authors. In *IFIP International Information Security Conference*, pages 339–350. Springer.

Richong Zhang, Zhiyuan Hu, Hongyu Guo, and Yongyi Mao. 2018. Syntax encoding with application in authorship attribution. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2742–2753, Brussels, Belgium. Association for Computational Linguistics.