Few-shot Authorship Attribution in English Reddit Posts

Anonymous ACL submission

Abstract

Authorship attribution (AA), an area of research seeking to identify the author of a particular text, is typically conducted on a closed set of authors, and often on certain forms of text, such as edited and less colloquial language like that available in news articles. This paper introduces a few-shot learning approach using prototypical networks and a mix of stylometric and pre-trained transformer-related features, as applied to Reddit data.

By employing few-shot learning and applying our efforts to social media text, we are looking to expand beyond the typical AA application– allowing for disjoint author sets and shorter, more colloquial forms of English. Additionally, using subreddit IDs as a proxy for topics, we explore cross-topic analysis and differentiate performance accordingly. In so doing, we test the limits of AA, with the goal of setting a baseline for performance and assessing viability of few-shot learning for this task. Of the exhibited models, those trained with transformer embeddings performed well compared to ones with only stylometric features, and accounting for differing subreddits showed varying performances across models.

1 Introduction

Authorship attribution (AA) is a natural language processing (NLP) task focused on identifying the author of a piece of text out of a small pool of potential authors. Real-world applications include plagiarism detection as well as forensic and historical/literary identity tracing (Meyer zu Eissen et al., 2007; Kestemont et al., 2016; van Cranenburgh, 2012).

While traditional classification approaches rely on having large quantities of text both for the unknown source as well as for each possible author, few-shot learning has been applied to a range of other classification tasks and shown success in reducing the amount of labeled data required (Wang et al., 2020; Tsimpoukelli et al., 2021; Geng et al., 2019). This paper focuses on AA via a dataset of Reddit posts and comments, i.e., the objective of the few shot learning task in this context is to distinguish between authors given a handful of example Reddit comments for each. Traditional techniques rely on stylometrics, features that capture how a person writes rather than the content they tend to write about (Stamatatos, 2009). Stylometrics can be subdivided into different types of features based on what the feature intends to encode: e.g., stylistic, lexical, syntactic, and character n-gram features. In the present work, various combinations of these feature sets, as well as more recent transformer embeddings, are used to identify which features are most suited to authorship attribution in the few-shot scenario.

In addition to identifying optimal feature sets, we also consider whether the inclusion of non-stylometric approaches, e.g. transformers, may be allowing the model to unintentionally focus on "topic" rather than authorship. For example, if two comments are written on the same subject matter but by two different authors, will the model incorrectly assume shared authorship? Stylometric approaches are intended to avoid this scenario but may have overall lower performance. Thus it is important to identify whether there is a trade-off between overall accuracy and errors based on subject matter.

2 Related Work

Historically, research into AA and related tasks often falls into one of two categories, classification and similarity-based models or systems (Stamatatos, 2009). Typical classification setups require a dataset with a closed set of authors, employing traditional supervised learning with the authors identified in the training set also appearing in the test set, though with new text samples and often a limited number of authors (Stamatatos,

2009). On the other side of the coin, similarity-based systems lend themselves to semi-supervised or few-sample scenarios, allowing for a disjoint and possibly larger set of potential authors. This work straddles the line between the two, using traditional few-shot learning methodology with a larger number of authors (1,000 - 10,000) in a closed set, while maintaining the few-shot framework of a small number of unseen authors given a small, defined number of sample texts per author.

2.1 Few-Shot Learning

The approach defined in this paper adheres to an *n-shot* training and evaluation process, a methodology employed in many few-shot learning setups, and conforms to typical n-shot terminology and episodic training (Vinyals et al., 2017). In this context, an *episode* is comparable to a batch and contains a support and query set; the framework randomly selects a specified number of classes and samples associated per class for a given {support, query} set–"k-way" will be used to describe the number of classes in a set; "n-shot" will be used to describe the number of samples supplied per class. Thus, 5-shot, 5-way classification will train based on sets of five authors with five sample texts apiece.

Though there are several networks that can be used with the episodic training process (Koch et al., 2015; Vinyals et al., 2017; Sung et al., 2018), the experiments presented in this paper will make use of a prototypical network (Snell et al., 2017). In this case, the classes and samples used in the support set are used to create a "prototype" of a class, an average of embeddings for each class specified in the support set. Data points from the query set are then mapped to a particular class based on minimum squared euclidean distance from the datapoint to the prototypes created with the support set (Snell et al., 2017).

2.2 Authorship Attribution

As mentioned above, many authorship identification-related studies rely on specific feature type generation, often subscribing to feature types thought to indicate writing style, or *stylometry* (Stamatatos, 2009; Ma et al., 2020). Common feature types in this field include character n-grams and syntactic and lexical-based information (Stamatatos, 2009; Sapkota et al., 2015; Ma et al., 2020). For example, function words such as determiners {the, a, an} and prepositions {upon, into, under}, use of punctuation and casing, and even average length of words can contribute to this feature extraction process. Note that this process is highly language dependent. Salient character n-grams and appropriate window size will likely vary by language, for example, and punctuation, casing, and even word length will differ in multilingual contexts. This paper restricts itself to English data, and the language-specific facets of the feature extraction process do affect which feature types are defined in the approach.

In attempting to define writing style, this methodology would separate style from topic, in some form or fashion, which is not always simple to implement. The objective in this case would be to identify authors in a cross-topic situation without inadvertently skewing the model towards identification via subject matter, especially by picking up topic-related tokens or n-gram features, a common approach in supervised text classification. Explicitly delineating a distinct style versus topic language is not cut and dried, and cross-topic author identification can be a difficult task (Murauer and Specht, 2021; Halvani et al., 2016). In fact, variations on topic masking or cross-topic classification have been explored to account for this concept, and define its impact on authorship attribution and identification tasks (Altakrori et al., 2021; Sari et al., 2018; Stamatatos, 2018).

This is especially notable given a potential real-world application, i.e., a model trained on text from an author writing about a particular subject matter that cannot identify the writings of the same author on a different topic. This can easily be imagined in the reverse–one author is mistaken for the other due to commonalities in subject matter. Cross-topic research can also be extended to genre and domain variations, though the work presented in this paper will focus on the cross-topic aspect (Barlas and Stamatatos, 2020) as applied to differing subreddits, or forums dedicated to specific topics, and authors' post history in these varying subreddits.

Along the same vein, authorship identification research has begun to incorporate pre-trained transformer models into new training approaches, though the potential for cross-topic performance variance is a concern that has been noted and is in the process of being addressed. Building on the past several years of producing state-of-the-art performance in multiple fields on natural language processing (NLP), it is no surprise transfer learning with pre-trained transformer models has been

extended to authorship identification. Indeed, recent research in this domain has made use of these breakthroughs (BERT, ELMO, etc.) (Devlin et al., 2019; Peters et al., 2018), with measured success (Fabien et al., 2020; Manolache et al., 2021; Murauer and Specht, 2021).

Generally, the application of these pre-trained models has sparked concern that performance of these models may be overly relying on topic-based cues, rather than style indicators (Manolache et al., 2021), and research into this area has suggested that stylometric-based models may provide more stability than their corresponding BERT-flavored models (Altakrori et al., 2021). Recent work in this area has shown promise, however–Fabien et al. were able to gain optimal performance in some contexts by combining stylometric and transformer-based features, and Manolache et al. mitigate this topic reliance by using disjoint training and evaluation sets (2020; 2021).

The work presented in this paper will build on this prior research, combining established stylometric features with more recent NLP transfer learning techniques. In order to push beyond "ideal" authorship attribution conditions, we present evaluations with an eye towards topic variation, as well as differing amounts of supplied training data. As AA is an NLP task presumably affected so largely by real-world variables, and with the potential for notable negative consequences, we intend to contribute to existing literature attempting to quantify and describe real-world performance in this area of research.

3 Data

The training and evaluation datasets were constructed using the `pushshift.io` API[1], extracting Reddit posts from 5,000 subreddits in December 2020. Pushshift uses Reddit's API[2] to collect and archive data from the Reddit platform to offer researchers more convenient access to such data (Baumgartner et al., 2020). Pushshift has provisions in place to give users the option to remove their data from the public-facing API[3].

From this larger dataset, the data used within these experiments was pulled to account for 10,000 authors posting at least 10 times within this time frame. When working with this data, unique authors and subreddits were recorded using the equivalent of a unique ID, rather than a name or handle.

Unless otherwise specified, models reported in this paper were trained using the training and validation splits available in Table 1, hereafter referred to as "general" models. Varying experiments call for different test sets depending on the analytical objective, but all evaluations adhere to disjoint training, validation, and test author sets, i.e., authors used in training are distinct from those used in validation, just as authors used in testing are distinct from those in both the training and validation sets.

Metric	Train	Val.	Test
No. Authors	8,000	1,000	1,000
Posts / Author (med.)	16	16	14
Subreddits / Author (med.)	4	5	4
Characters / Post (med.)	256	277	262
Tokens / Post (med.)	52	56	54

Table 1: Relevant statistics of the general dataset used in our experiments. Most models reported in this work were trained using the above training and validation splits. "Med." specifies the median, e.g., "Posts / Author (med.)" denotes the median number of samples per author in the indicated split.

4 Approach

This work includes a series of experiments with trained models using a variety of feature types, defined in Table 2. For ease of discussion, the feature sets included are referenced using the corresponding code, such as $S1$, $S2$, etc., and this notation is maintained throughout the paper. In this context, feature sets $S1$, $S2$, $S3$, and $S4$ are considered stylometric feature types, and T describes transformer model embeddings, extracted via a BERT-flavored model (DistilBERT) available through the Hugging Face Hub[4] and transformers library[5]. (Sanh et al., 2019).

All part-of-speech (POS) tag features are extracted using the ARK Tweet NLP POS Tagger[6], designed with social media text in mind (Gimpel et al., 2010; Owoputi et al., 2012), and tokenization-based features rely on NLTK's TweetTokenizer[7]. Tables included in this paper will include an addi-

[1] https://github.com/pushshift/api
[2] https://www.reddit.com/wiki/api-terms
[3] https://www.reddit.com/r/pushshift
[4] https://huggingface.co/models
[5] v. 4.15.0; https://github.com/huggingface/transformers
[6] http://www.cs.cmu.edu/ ark/TweetNLP
[7] v. 3.6.7; https://www.nltk.org/api/nltk.tokenize.casual.html

tional horizontal line splitting reported scores into two groups. This is used to easily distinguish between models trained on only stylometric features, and those that include transformer embeddings.

Stylometric features are pre-calculated using the entirety of the training dataset, rather than only those available in a given support set. This applies primarily to n-grams, extracted via characters or POS tags. Transformer-based embeddings rely on a given pre-trained model, specifically `distilbert-base-uncased`[8]. Though not reported in this paper, other transformer models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) were examined for performance, but more extensive evaluations were ultimately performed using DistilBERT due to its reliable performance and fewer parameters (66M) (Sanh et al., 2019). Other models were not necessarily less successful than DistilBERT, rather DistilBERT's GPU memory footprint was preferred while maintaining quality performance.

Once features have been computed, feature sets are then concatenated using a mid-fusion technique wherein each feature set is separately encoded and then concatenated to produce a single representation of the datapoint, as depicted in Figure 1. This concatenated representation is then used to train a prototypical network. These models are then applied in multiple contexts, to better describe general performance.

Training was conducted on an NVIDIA A100 GPU, with training times varying from under one hour to approximately one day based on feature types. Reported train times are based on a maximum of 300 iterations with 500 episodes each. Model weights were taken from the iteration with lowest validation loss, and higher scoring models tended to hit that target fairly early in the training process, 20 to 100 iterations. Smaller, stylometric models tended to skew towards faster training times, while models trained with DistilBERT embeddings skewed slower.

5 Results

The following experiments have been broken down into three different subsections: Categorized by Feature Type, Increasing N-shot, and Cross-Topic Analysis. Categorized by Feature Type analyzes the performances of models trained on a variety of concatenated feature types in a 5-shot, 5-way scenario, while Increasing N-shot charts model performance with increasing number of datapoints in the support set, 1- to 20-shot. The final subsection looks at varying performance given the impact of additional factors, namely topics within support and query sets.

5.1 Categorized by Feature Type

Given sets of differing feature types, how do models trained on stylometric vs. transformer-based features compare? Models and train/validation/test splits used here are hereafter referred to as *general* models and datasets. These models will be referred to again in other capacities as we discuss experimental runs and analyses in other sections.

Feature Codes	Val.	Test
S1	63.8	64.9
S2	40.7	42.4
S3	62.6	63.1
S4	56.5	57.0
S1 + S2	64.0	66.1
S1 + S2 + S3	69.4	70.3
S1 + S2 + S3 + S4	71.8	72.8
T	**87.7**	**87.7**
S1 + S2 + S3 + S4 + T	**87.6**	**88.0**

Table 3: Accuracy for validation and test sets: 5-shot, 5-way.

5.2 Increasing N-shot

To investigate the effect of varying numbers of samples per class, a separate dataset was derived from the December 2020 data. This dataset is comparatively reduced in size due to the requirements for additional posts per author. In order to chart results based on a support set with a range of 1 to 20 datapoints per author, the dataset was restricted to those authors with at least 25 posts or comments, leaving at least 5 texts for the query set. Table 4 records statistics for this dataset.

[8] https://huggingface.co/distilbert-base-uncased

Feature Set	Feature Code	Description
Stylistic	S1	Stylistic features including normalized counts of digits, alphabetic characters, punctuation and special characters, vocabulary complexity, and more
Lexical	S2	Normalized counts of 6 part of speech (POS) tags describing function words
Syntactic	S3	Counts of raw POS tags and POS tag n-grams (bigrams and trigrams)
Character N-grams	S4	Counts of character n-grams (bigrams and trigrams)
Transformer	T	Transformer model embeddings (DistilBERT)

Table 2: Describes each of the five feature sets used in various models trained for this task. Feature codes denoted above are used to easily reference these feature sets throughout the paper.

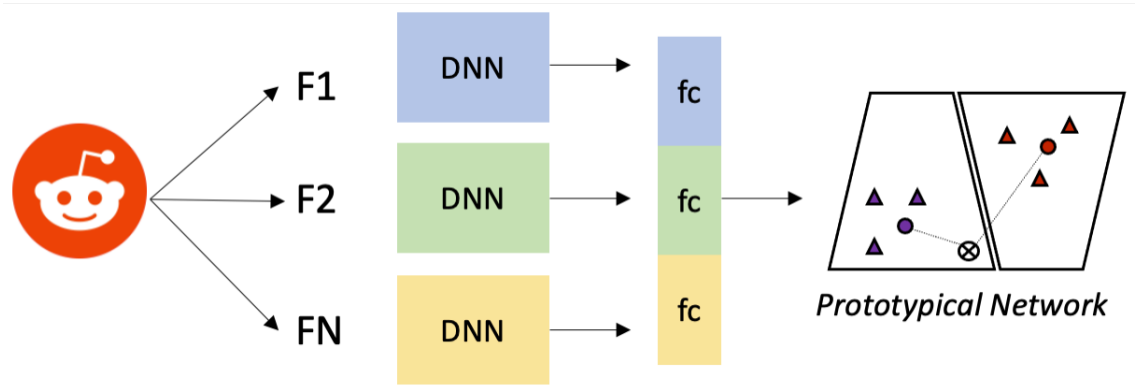

Figure 1: Rendering of model architecture using mid-fusion. F1,...FN represent feature types such as syntactic features, character n-grams, transformer embeddings, etc. Please note we use F and f to refer to any type of feature, stylometric (S) or transformer (T). These are encoded using deep neural networks, concatenated, and trained using a prototypical network. The prototypical network image was created based on a figure included in Snell et al. (2017).

Metric	Train	Val.	Test
No. Authors	800	100	100
Posts / Author (med.)	40	41	35.5
Subreddits / Author (med.)	5	4	5
Characters / Post (med.)	270	191	240
Tokens / Post (med.)	52	39	50

Table 4: Train and validation splits differ for these models (and only these models) due to the nature of the experiment.

Using this dataset, models were trained in a 5-shot, 5-way scenario, with the number of datapoints increasing between 1 and 20 for subsequent testing. When running evaluations, support sets were constructed using a [1-20]-shot 5-way paradigm. Figure 2 shows a similar trend of performance increase across model types, with a sharp performance increase until about 4-shot and then a gradual increase to a plateau.

5.3 Cross-Topic Analysis

This section focuses on the cross-topic aspect of authorship attribution, using subreddit IDs as stand-ins for various topics shared or excluded across authors. As discussed above, cross-topic authorship identification efforts often report lower performance values, which can adversely harm practical applications. To further examine this element of AA, this section breaks the following relevant experiments into two parts, Performance by Author "Difficulty" and Approximating Disjoint Topic Coverage. These analyses attempt to quantify an element of cross-topic analysis by aligning identified (anonymized) authors with recorded subreddits associated with specific posts.

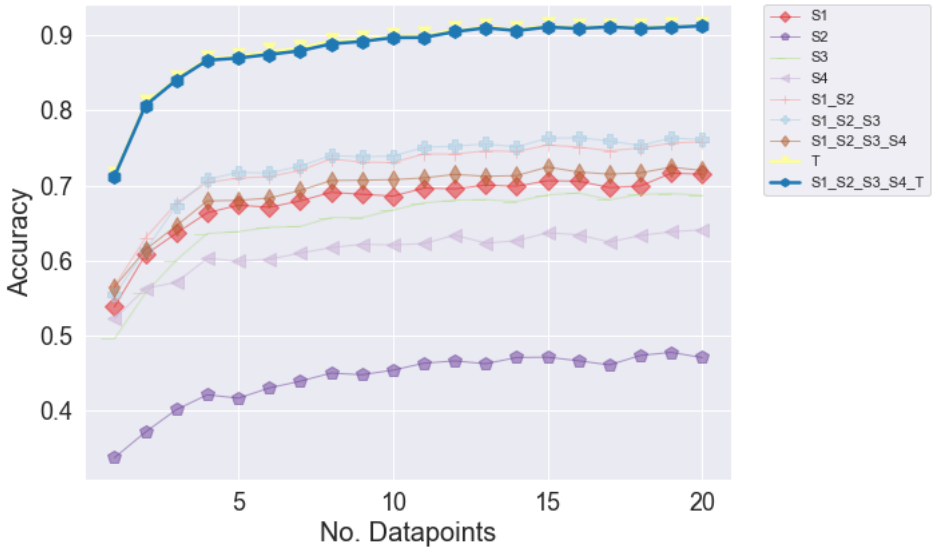

Figure 2: Chart of model performance based on a range of datapoints per author (1-20) for a given support set.

5.3.1 Performance by Author "Difficulty"

In this section, there are two main datasets to make note of, described in Table 5. Each one is a subset of the original test set described in Table 1, dividing authors into "easy" and "hard" sets of 200 authors each and corresponding posts. These authors were selected based on the number of unique subreddits in which they posted, assuming posts within the same subreddit likely focused on a particular topic. Thus, authors posting in fewer subreddits were deemed "easy," and those posting across more subreddits were deemed "hard."

Dataset	Subreddits / Author (med.)
Easy	1
Hard	10

Table 5: A total of 400 authors extracted from the original test data described in Data. Easy authors have the fewest number of unique subreddits; hard authors have the most.

Table 6 shows the results of running models on the Easy/Hard datasets. As expected, the "Easy" dataset shows higher accuracy across the board when compared to the "Hard" dataset, presumably due to less varied topics within the identified posts.

Feature Codes	Easy	Hard
S1	77.2	58.5
S2	53.6	35.9
S3	76.6	55.7
S4	70.3	49.1
S1 + S2	78.2	60.0
S1 + S2 + S3	81.5	65.3
S1 + S2 + S3 + S4	82.9	66.6
T	94.8	83.4
S1 + S2 + S3 + S4 + T	94.7	83.4

Table 6: General models applied to a subset of the initial test set outlined in Categorized by Feature Type; *easy* and *hard* authors are identified based on the number of subreddits accounted for in corresponding texts.

5.3.2 Approximating Disjoint Topic Coverage

Rather than focusing on overall topic variability by author, the following analysis concentrates instead on an approximation of disjoint topics (subreddits) between the support and query sets for a given author. Maintaining completely disjoint authors between training, validation, and test sets, this particular effort ensures at least 50 or 100% of the query set includes posts from subreddits not included in the support set when evaluating the model. As this process requires differing subreddits between support and query sets but does not maintain disjoint sets of subreddits between train/validation/test splits, we have used the term *approximating* disjoint topic coverage in creating these cross-topic

sets. As many authors in the general test set did not have the required variation in subreddit postings, separate test sets were curated from the December 2020 source dataset. Specifics are available in Table 7.

Metric	D50	D100
No. Authors	857	105
Posts / Author (med.)	32	52
Subreddits / Author (med.)	13	12
Characters / Post (med.)	300	342
Tokens / Post (med.)	62	67

Table 7: D50 and D100 are abbreviations for Disjoint-{50, 100}. 50 and 100 refer to the percentage of minimum topic distinction between support and query sets for a given class.

Table 8 reports findings, comparing "disjoint" performance from the "general" test set used in other sections of this paper. Due to the challenge of the task, a drop in performance is expected when comparing to the general models, and we do indeed see that in this case. Additionally, Disjoint-100 generally performs better than Disjoint-50, which may seem counter-intuitive, given the guaranteed disjoint subreddits between support and query sets. A possible explanation would suggest curating support and query sets along topic lines can actually improve cross-topic performance, presumably due to varied topic exposure.

Feature Codes	Gen.	D50	D100
S1	64.9	58.5	62.6
S2	42.4	35.4	40.8
S3	63.1	55.1	61.0
S4	57.0	49.8	54.1
S1 + S2	66.1	58.8	62.6
S1 + S2 + S3	70.3	63.5	66.1
S1 + S2 + S3 + S4	72.8	65.4	67.8
T	87.7	81.1	82.2
S1 + S2 + S3 + S4 + T	88.0	81.0	82.7

Table 8: Results of cross-topic support and query sets as compared to the general, random test set used in Table 3. Applied models are the same as those outlined above, *general* models.

6 Conclusion

Models using transformer embeddings perform well overall, compared to the outlined stylometric feature sets. Similar patterns in performance fluctuation, i.e., cross-topic and increasing n-shot analyses, appear to apply across different feature types. In general, the 5-shot, 5-way models appear to rapidly increase in accuracy up to 4 or 4 datapoints, with gradual increases beyond that benchmark. Additionally, cross-topic analysis suggests performance is heavily influenced by topic variation, whether in variation available per author overall or via curated support and query sets. Additional experimentation looking into disjoint support and query sets, as well as train/validation/test splits, may yield more information given some of the results reported here.

6.1 Limitations and Future Work

The Reddit dataset compiled for these experiments is from a compressed time frame, is focused only on English-language posts, and analysis was conducted without nuanced regard for the specific types of topics included in the varying subreddits. It is possible some of the included subreddits are more semantically related to each other than others. Future work could curate datasets with specific attention to subreddits, and perhaps use similarity measures to describe or differentiate topics.

Regarding feature fusion techniques, early iterations of models did pursue early fusion, but more rigorous experimentation may have caught more effective techniques. Perhaps there is more work that can be done here beyond strict early versus mid-fusion; certain combinations of stylometric feature types could potentially benefit from earlier concatenation. Likewise, more rigorous exploration of stylometric features may have unknown benefits to those models effected. Subtle variations in features, such as feature reduction techniques, could have an impact. This was investigated to an extent, but future work could concentrate specifically on these features (and other stylometric feature types).

6.2 Ethical Considerations

Data privacy can be considered one the fundamental ethical concerns in the field of machine learning and artificial intelligence research. Given this position, any work extracting author or user identities needs to be carefully approached. In forensic or law enforcement applications in particular, the lim-

itations of real-world applications must be taken into account considering real-world consequences, as well as data privacy issues and potential for misuse (Solove, 2007). The same techniques used to expose the author of a threatening letter can also be used by oppressive governments (and private entities) to target individuals belonging to marginalized groups, for example.

Beyond this aspect, there is also a legitimate concern regarding de-anonymization of publicly available articles and social media postings, which has contributed to studies into automatic anonymization or author masking (Brennan et al., 2012; Emmery et al., 2021; Bo et al., 2021). Demasking the author of an anonymous posting beyond law enforcement/prosecution can have devastating implications, including career ramifications and loss of public anonymity, which can lead to public allegations and harassment (Ainsworth and Juola, 2019). These risks must be considered when approaching author identification tasks.

References

Janet Ainsworth and Patrick Juola. 2019. Who Wrote This?: Modern Forensic Authorship Analysis as a Model for Valid Forensic Science. 96:30.

Malik H. Altakrori, Jackie Chi Kit Cheung, and Benjamin C. M. Fung. 2021. The Topic Confusion Task: A Novel Scenario for Authorship Attribution. *arXiv:2104.08530 [cs]*. ArXiv: 2104.08530.

Georgios Barlas and Efstathios Stamatatos. 2020. Cross-Domain Authorship Attribution Using Pre-trained Language Models. In Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis, editors, *Artificial Intelligence Applications and Innovations*, volume 583, pages 255–266. Springer International Publishing, Cham. Series Title: IFIP Advances in Information and Communication Technology.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The Pushshift Reddit Dataset. *arXiv:2001.08435 [cs]*. ArXiv: 2001.08435.

Haohan Bo, Steven H. H. Ding, Benjamin C. M. Fung, and Farkhund Iqbal. 2021. ER-AE: Differentially Private Text Generation for Authorship Anonymization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3997–4007, Online. Association for Computational Linguistics.

Michael Brennan, Sadia Afroz, and Rachel Greenstadt. 2012. Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity. *ACM Transactions on Information and System Security*, 15(3):1–22.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Chris Emmery, Ákos Kádár, and Grzegorz Chrupała. 2021. Adversarial Stylometry in the Wild: Transferable Lexical Substitution Attacks on Author Profiling. *arXiv:2101.11310 [cs]*. ArXiv: 2101.11310.

Mael Fabien, Esau Villatoro-Tello, Petr Motlicek, and Shantipriya Parida. 2020. BertAA: BERT fine-tuning for Authorship Attribution. page 11.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. 2019. Induction networks for few-shot text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3904–3913, Hong Kong, China. Association for Computational Linguistics.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2010. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments:. Technical report, Defense Technical Information Center, Fort Belvoir, VA.

Oren Halvani, Christian Winter, and Anika Pflug. 2016. Authorship verification for different languages, genres and topics. *Digital Investigation*, 16:S33–S43.

Mike Kestemont, Justin Stover, Moshe Koppel, Folgert Karsdorp, and Walter Daelemans. 2016. Authenticating the writings of Julius Caesar. *Expert Systems with Applications*, 63:86–96.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese Neural Networks for One-shot Image Recognition. page 8.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Weicheng Ma, Ruibo Liu, Lili Wang, and Soroush Vosoughi. 2020. Towards Improved Model Design for Authorship Identification: A Survey on Writing Style Understanding. *arXiv:2009.14445 [cs]*. ArXiv: 2009.14445.

Andrei Manolache, Florin Brad, Elena Burceanu, Antonio Barbalau, Radu Ionescu, and Marius Popescu. 2021. Transferring BERT-like Transformers' Knowledge for Authorship Verification. *arXiv:2112.05125 [cs]*. ArXiv: 2112.05125.

Sven Meyer zu Eissen, Benno Stein, and Marion Kulig. 2007. Plagiarism Detection Without Reference Collections. In *Advances in Data Analysis*, pages 359–366, Berlin, Heidelberg. Springer Berlin Heidelberg.

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.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, and Nathan Schneider. 2012. Part-of-Speech Tagging for Twitter: Word Clusters and Other Advances. page 15.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Upendra Sapkota, Steven Bethard, Manuel Montes, and Thamar Solorio. 2015. Not All Character N-grams Are Created Equal: A Study in Authorship Attribution. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–102, Denver, Colorado. Association for Computational Linguistics.

Yunita Sari, Mark Stevenson, and Andreas Vlachos. 2018. Topic or Style? Exploring the Most Useful Features for Authorship Attribution. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 343–353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical Networks for Few-shot Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Daniel J. Solove. 2007. 'I've Got Nothing to Hide' and Other Misunderstandings of Privacy. SSRN Scholarly Paper ID 998565, Social Science Research Network, Rochester, NY.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asi.21001.

Efstathios Stamatatos. 2018. Masking topic-related information to enhance authorship attribution. *Journal of the Association for Information Science and Technology*, 69(3):461–473. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asi.23968.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. 2018. Learning to Compare: Relation Network for Few-Shot Learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, Salt Lake City, UT. IEEE.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34.

Andreas van Cranenburgh. 2012. Literary authorship attribution with phrase-structure fragments. In *Proceedings of the NAACL-HLT 2012 Workshop on Computational Linguistics for Literature*, pages 59–63, Montréal, Canada. Association for Computational Linguistics.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2017. Matching Networks for One Shot Learning. *arXiv:1606.04080 [cs, stat]*. ArXiv: 1606.04080.

Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)*, 53(3):1–34.