ATLANTIS PRESS
Atlantis Studies in Uncertainty Modelling, volume 3
Joint Proceedings of the 19th World Congress of the International Fuzzy Systems Association (IFSA), the 12th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT), and the 11th International Summer School on Aggregation Operators (AGOP)

A Sentiment-Based Author Verification Model Against Social Media Fraud

Khodor Hammoud[a] and **Salima Benbernou**[b] and **Mourad Ouziri**[c]
[a]LIPADE, Université de Paris, France, khodor.hammoud@etu.u-paris.fr
[b]LIPADE, Université de Paris, France, salima.benbernou@u-paris.fr
[c]LIPADE, Université de Paris, France, mourad.ouziri@u-paris.fr

Abstract

The widespread and capability of Iot devices have made them a primary enabler for online fraud and fake authorship on social media. We present a novel approach, which uses sentiment analysis, to solve the problem of author verification in short text. We perform experimentation with our model on tweets, and show that it yields promising results.

Keywords: authorship verification, machine learning, sentiment analysis, short text, keyword search.

1 Introduction

Internet of things (IoT) is the concept of having a widespread infrastructure of inter-connected devices capable of communicating over the internet. IoT devices are capable of performing various computational tasks, and have the ability to work in organized clusters. The cheap manufacturing cost, combined with their high portability have made IoT devices available in high volumes, where estimates say that there are 35 billion connected IoT devices today. Thus, important security risks are raised as any device could be used as an attack channel [10]. The widespread utility of IoT has made it one of the bigger enablers of online malicious attacks, specifically in the domain of spreading fake news [11]. If we combine that fact with the widespread of online social media, we get a system capable of spreading fake information to people's social networks. IoT botnets, which are capable of producing artificial social activity, fake information, and can impersonate others' online presence, have facilitated the creation and widespread of social fraud techniques, like social media fraud (SMF) [23], online fake news [3], and fake online authorship [27].

The importance of online social networks in today's daily life gave rise to online influencers whose opinions have an effect on people's decisions. These influencers make a primary target for IoT fraud botnets, where these botnets can impersonate an influencer and post misleading information under her identity [2, 26]. The problem of identifying if a piece of information is truly published by a claiming author is called Author Verification (AV). AV frames the question of the true authorship of a piece of document as a classification problem: given an author A, a set of documents D authored by A, and a document d, determine whether d was authored by A. The vast majority of research about AV has been dedicated towards finding the author of long texts. However, with today's reliance on fast and short messages for communication, there is a need for AV on short text more than ever. More specifically, there is a need for AV on micro-messages, like tweets. Previous work has shown that it is difficult to maintain good performance when an AV system works on long text vs shorter text [22]. However, there has been many recent AV projects that experimented with short web data such as emails [1], web forum messages [30] and blogs [19].

Current approaches for AV utilize a variety of techniques with a final goal of detecting the author's writing profile. They work by extracting stylistic features from the text, then finding a writing pattern to uniquely identify the author. These features include word n-grams [25], character n-gram [28, 15, 29], vocabulary richness [1] and part-of-speech n-grams [21]. These approaches have been applied for both long and short text, with varying degrees of success.

However, the problem with such approaches is that they rely heavily on the structure the text is written in. But micro-messages don't provide enough data to detect a stylistic pattern that can uniquely identify an author. In addition, studies have shown that average word length is neither stable within a single author, nor does it necessarily distinguish between authors [14].

In this paper, we focus on the problem of Author Verification (AV) [31] in social media. More specifically, we experiment with AV on twitter tweets, although our approach can be applied to any form of short text. We make the following contributions:

- We studied in depth existing algorithms for author verification dedicated for long text.
- We experiment with one of the top performing algorithms of the Pan 15 AV task [32], which caters for one of the most important benchmarks to which new AV approaches refer and compare against, and show that this performs poorly when handling short text.
- We propose a novel approach for AV, which is a sentiment based author verification method for short text. To the best of our knowledge, we are the first to utilize a sentiment-based technique for AV.
- We experiment with our sentiment-based author verifier on a tweets dataset, and analyze the results and performance.

The rest of the paper is outlined as follows: in section 2 we provide a motivating example, in section 3 we give an overview of existing work, in section 4 we present our sentiment-based author verification model, in section 5 the experimental studies and results are presented, we discuss the results in section 6, and conclude in section 7.

2 Motivating Example

Consider a VIP who‘s opinion can affect peoples decisions, like a politician or an influencer. This type of person would make a great target for imposters to try and impersonate with the aim of spreading fake news through a network using an IoT device. A statement issued by this VIP can go as much as to affect the stock markets, like the rise in Signal's stock market value because of a tweet from Elon Musk recommending people to switch from using Whatsapp to using Signal [1]. Current style-based AV approaches suffer in performance when working with short text. However, when we consider the ability to imitate the author's way of writing, it might become impossible for those approaches to detect fake authorship.Consider the following quote from Elon Musk[2]:

> "*You can't have a person driving a two-ton death machine.*"

If we simply remove 2 characters from the above quote, so that "can't" becomes "can", the same writing style would be maintained. An algorithm that is trained to detect Elon Musks‘s style of writing would still recognize the quote:

> "*You **can** have a person driving a two-ton death machine.*"

as an Elon Musk quote. These kinds of edits would go by unnoticed by style-based AV systems since the doctored fake statement is fabricated following the structure of an authentic one, even though the meaning is completely changed.

We argue that for an AV system to work well with short text, it needs to go beyond style-based features and focus more on semantics. A good AV system should be able to not only detect an author‘s style of writing, but also have an understanding of their general opinions towards specific topics.

3 Related Work

Boenninghoff et al. [6] propose ADHOMINEM, a new attention-based neural network topology for similarity learning in short text. The network applies characters-to-word, words-to-sentence and sentences-to-document encoding to extract document-specific features to make a similarity analysis between two documents. They apply their model to a dataset of amazon reviews which they develop themselves.

Buddha Reddy et al, [8] provide a new approach for authorship verification using a non-uniform distributed term weight measure to calculate term weights in the text instead of TF-IDF. Castro et al. [9] propose a solution for AV based on comparing the average similarity of an unknown authorship text with the Average Group Similarity between text samples from the target author. They also performed experiments with a total of 17 types of linguistic features, grouped int 3 categories: character, word and syntactic, and used six similarity functions. Halvani et al. [13] propose an AV approach that considers only topic-agnostic features in its classification decision.

The PAN series of shared tasks is considered one of the most important benchmarks and references for authorship attribution research. The PAN authorship verification tasks [17, 33, 32, 4] tackle what Koppel et al. [20] called the "fundamental problem" in authorship attribution: Given two documents, are they written by the same author? Bevendorff et al. [5] review the PAN authorship verification task and state that the experiment

[1] https://www.cnbc.com/2021/01/11/signal-advance-jumps-another-438percent-after-elon-musk-fueled-buying-frenzy.html

[2] https://www.youtube.com/watch?v=Q6q-HZBLy0I

Algorithm 1 Opinion History Creation

Input: D: Documents of known authorship
Output: H: Opinion history

procedure CREATEOPINIONHISTORY
  Foreach $d_i \in D$:
    $kp_i = extract\_keyphrase(d_i)$
    $we_i = extract\_word\_embeddings(kp_i)$
    $s_i = infer\_sentiment(d_i)$
    $H.append((kp_i, we_i, s_i))$
  End Foreach
  return H
end procedure

Algorithm 2 Authorship Prediction

Input: d_x: Document of unknown authorship
  H : Opinion history from Algorithm 1
Output: *isAuthor*: Is A the author, with confidence

procedure ISAUTHOR
  $kp_x = extract\_keyphrase(d_x)$
  $we_x = extract\_word\_embeddings(kp_x)$
  $s_x = infer\_sentiment(d_x)$
  $topn = get\_top\_n\_similar(kp_x, H)$
  $confidence = weighted\_average(topn)$
  $\tau = calc\_threshold(topn)$
  $isAuthor = is\_within(s_x, \tau)$
  return $(is\_author,\ confidence)$
end procedure

design presented at PAN may not yield progression of the state of the art. They tested what they call a "Basic and Fairly Flawed" authorship verifier model which performs competitively with the best approaches submitted to PAN until that time, which were the PAN 2013 [17], 2014 [33] and 2015 [32] AV tasks.

Dam et al. [34] investigate the influence of topic and time on AV accuracy. Regarding topic influence, they found that cases with documents of similar topics overall (positive and negative) were found to increase accuracy of AV. As for the influence of time, they found that writing style indeed changes over time, by comparing Wikipedia Talkpages contributions made within a week with Wikipedia Talkpages contributions made years apart. AV is more accurate when comparing texts that have been written within a short period of time.

4 Sentiment-Based Verifier

4.1 Framework Overview

We base our approach on the idea of analyzing people's opinions towards certain topics. If an author A has a well-known history of liking Apple phones, for example, then her suddenly publishing a negative statement about Apple phones would be unlikely. We can use this knowledge to create a system that checks for inconsistencies between the history of opinions belonging to an author A about a certain topic T, and a recent opinion towards T, and deduce the likelihood of this recent opinion being authored by A.

This approach can be applied to short texts, long texts, social media posts, and even excerpts of quotes. As long as a piece of text is known to belong to A, it can be divided into individual sentences, and the associated keyphrase/opinion pair can be extracted and added to the opinion history we have about A. A visualization of the overview of our model can be seen in Figure 1.

4.2 The Process

Step 1: keyphrase extraction and embedding. A keyphrase is the main topic that a piece of text revolves around. For example, the keyphrase of the sentence *"I like to use Apple phones"* is *Apple phones*. Given an Author A suspected of publishing a piece of text d_x, we collect text previously authored by A, and then extract the keyphrases. A keyphrase acts as the main topic that the text is centered around. For every extracted keyphrase kp_i, we calculate its word embeddings we_i. This allows us to do similarity search between keyphrases in a later stage. In the scenario where our text of unknown authorship centers around a keyphrase kp_x that has not been precisely encountered by our model before, we can estimate it by looking for the keyphrases most similar to kp_x. Word embeddings, combined with the sentiments extracted in step 2, form what we are calling the opinion history that is used for authorship inference. Algorithm 1 presents the opinion history creation process. By using word embeddings, running a similarity search algorithm, like cosine similarity, between the word embeddings of every keyphrase in the opinion history, and the word embeddings of kp_x returns the keyphrases that are most semantically similar to kp_x. The similarity score between kp_x and the most similar keyphrases can then be used as the system's confidence score.

Step 2: sentiment extraction. For every text that we've extracted the keyphrase of, we also extract the text sentiment.This way, the combination of the keyphrase/sentiment tells us what the author's opinion towards the topic of the text is. Sentiments extracted are in the form of a continuous value between -1 and 1, where the closer the value is to 1, the more positive the sentiment is, the closer it is to -1, the more negative the sentiment is, and the closer the value is to 0, the more neutral it is.

Step 3: application on the unknown text. The final

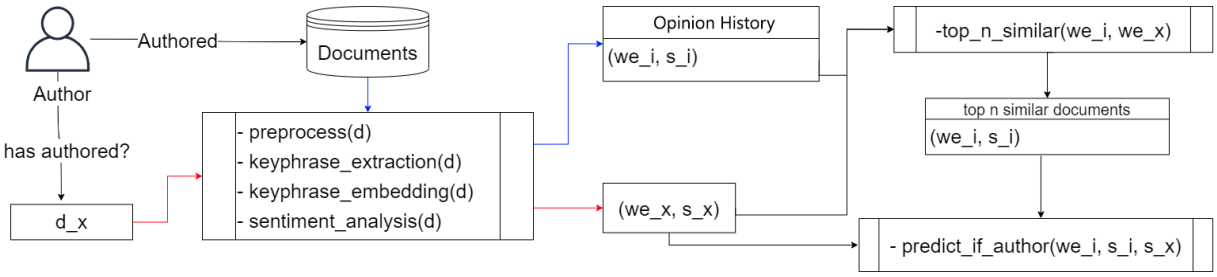

Figure 1: Sentiment-based verifier architecture

step of our model is to process a new text d_x of questioned authorship. After creating the opinion history of the suspected author A, we apply a similar procedure to d_x as the one we applied to every document when constructing the opinion history. We extract the keyphrase of d_x and calculate its word embedding we_x, and extract the sentiment s_x of d_x. Then, we run cosine similarity between we_x and all the keyphrase embeddings in the opinion history to get the top n most similar keyphrases, which would reflect the top n most similar texts to d_x, in terms of context, which are published by A. Based on the sentiment of the top n most similar texts, we can predict what should the sentiment of d_x be. That can be calculated as an average weighted sum:

$$sp_x = \frac{\sum_{i=1}^{n} cos\_sim(we_i, we_x) \cdot s_i}{n} \tag{1}$$

where sp_x represents the predicted sentiment of d_x, cos_sim is the cosine similarity function, and we_i and s_i represent the word embedding and sentiment of text i in the top n most similar texts respectively. This predicted sentiment can then be compared with s_x, the real extracted sentiment of d_x, and based on a similarity threshold, the model can determine, with a certain confidence, if d_x was authored by A. Algorithm 2 presents the authorship prediction process.

4.3 Confidence and Threshold

Confidence. The confidence of our model is an estimate of the its performance when predicting whether A is the true author of a document of questioned authorship d_x. Calculating the confidence is dependent on the similarity between the word embeddings of each of the top n similar keyphrases we_i in the opinion history and we_x, the word embedding of the keyphrase of d_x, which is the document of questioned authorship. The higher the similarity between these embeddings (reflected by a cosine similarity value closer to 1), the higher is the confidence of the model's prediction. The confidence is calculated as the average of the similarities between we_x and each we_i. This decision can be justified as follows: the closer we_x is to each we_i, the closer the topic of d_x is to the topics of previous statements by A. Hence, the more likely sentiment s_x is to be consistent with the sentiments s_i of the top n similar documents.

Threshold. The sentiment of an author might vary significantly with regards to the same topic; she might be consistent or inconsistent with her opinions. In a scenario where she is consistent, she might always have a relatively positive opinion about Apple phones for example, i.e: $0.7 \leq s_i \leq 1 \ for \ i \in [1,n]$. While in an inconsistent scenario, her opinion might vary significantly from one tweet to another, i.e: her top 5 similar tweets about Apple phones can have the sentiments: $[-0.7,\ 0.3,\ 0.8,\ 0,\ -0.4]$. This adds an additional layer of difficulty, because sp_x would be averaging a wide range of values, which would not be a good representative of the author's opinion. In the consistent scenario, sp_x would be a good value to compare s_x to, since sp_x might be a value close to 0.8, so provided a certain *threshold* τ, we would just need to check if $s_x \in [sp_x - \tau,\ sp_x + \tau]$.

This problem can be solved by making the *threshold* adaptive to the spread of the sentiments s_i. We accomplish that by using the standard deviation of the sentiments (equation 2), where μ is the mean of the sentiments. The sentiment value is between -1 and 1. Thus, the value of σ is between 0 and 1. A σ value of 0 indicates that the sentiments are all equal; 0 spread.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_i - \mu)^2} \tag{2}$$

And a σ value of 1 indicates maximum spread, that is: sentiments are equally divided between the exact values -1 and 1. Using this information, the threshold can be calculated as σ plus a leniency parameter α which can be tuned to suit different use cases: $\tau = \sigma + \alpha$. This provides a way to adapt our threshold to the spread of the sentiments.

5 Experimental Studies

5.1 Implementation Details

For extracting keyphrases, we use KeyBERT [12], which is a keyword extraction technique that utilises BERT embeddings [35] to create keywords and keyphrases that are most similar to a document.

For calculating the word embeddings of keyphrases, we use Facebook's Fasttext [7] with the pre-trained English embeddings in dimension 300[3].

For sentiment analysis, we use VADER Sentiment Analysis [16], a lexicon and rule-based sentiment analysis tool that specializes in inferring sentiments expressed in social media.

5.2 Dataset

Long text dataset. For testing existing approaches, we use the PAN 15 author identificaiton task dataset, which is a training corpus composed of a set of problems, where each problem is described as a set of documents (1-10 per problem) belonging to a single known author, and exactly one document of questioned authorship. Within each problem. Each document lengths vary from a few hundred to a few thousand words.

Short text dataset. To evaluate our approach, we use the dataset developed by Schwartz et al. [28] containing ∼9,000 Twitter users with up to 1,000 tweets each. We depart from the preprocessing followed by Schwartz, since we pereserve dates, times and references to other users (@<*user*>). We do this since in our case, we are interested in the author's opinion towards different keywords mentioned in their tweets. For example, if the main focus of a tweet is to talk negatively about someone, the author is likely to mention that someone's twitter user name, and that username is likely to be detected as the tweet's keyword. Or, the author might, for example, like a specific year model of a car, but dislike one from another year. However, we do remove the @ sign from the beginning of the mention (@<*user*> becomes <*user*>), and we also replace web addresses with the meta tag *URL* as we don't see a contribution of such data to an author's opinion.

5.3 Metrics

For evaluating our model's performance, *recall*, *precision* and *F-1 Score* are used:

$$Recall = \frac{\#correct\_answers}{\#problems} \tag{3}$$

$$Precision = \frac{\#correct\_answers}{\#answers} \tag{4}$$

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

5.4 Experimentation Results

5.4.1 SPATIUM-L1

SPATIUM-L1[4] is an unsupervised authorship verification model developed by Kocher and Savoy [18]. It was submitted to the PAN at CLEF 2015 Author Identification task, and placed 4th in the evaluation on English language. This approach is based on simple feature extraction and distance metric, where the top k most frequent terms are selected to verify the validity of the author. This model performed well with an **AUC** score of 0.738 and a **c@1** [24] score of **0.689**. We run our experiments on this algorithm and the results are displayed in table 1.

Replication on long text. We have confirmed the results from the evaluation by testing the SPATIUM-L1 model on the PAN 15 AV dataset, and have gotten a **c@1** score of **0.6436** which is very close to the one reported at the PAN 15 evaluation of **0.689**.

Experimentation on short text. We run the SPATIUM-L1 algorithm on out short-text dataset. We vary the number of tweets per author used to infer the authorship of the tweet of unknown authorship, and we note that regardless of the number of tweets per account, the percentage of unknown outputs is high in comparison with that of the PAN 15 dataset. In addition, as the number of tweets per account increase, the **c@1** score decreases dramatically, and the number of unknowns increase greatly as well. This seams as a counter-intuitive behavior since one would expect better performance once more data is provided per account.

5.4.2 Sentiment-Based Verifier

We run our model on the short text dataset. We use the tweets of 3000 authors, each having 1000 tweets. For each author, we use 910 tweets to build the opinion history, and create a validation set composed of 20 tweets: 10 belonging to the same author, and 10 randomly selected from other authors. For every author, we run the model and do predictions on the respective validation

[3] https://fasttext.cc/docs/en/crawl-vectors.html

[4] https://github.com/pan-webis-de/kocher16

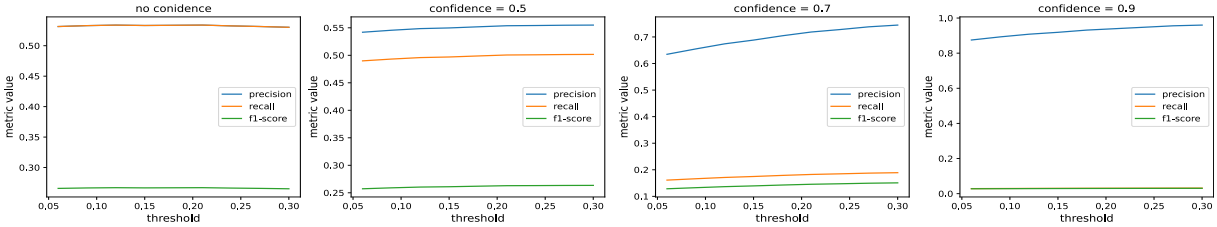

Figure 2: Model performance as a function of *threshold* with varying confidence

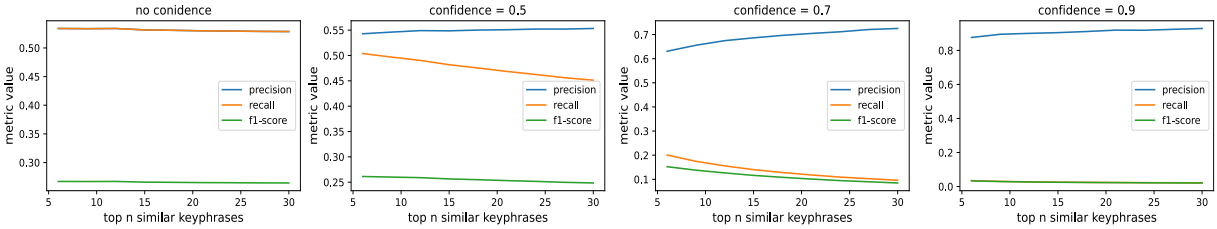

Figure 3: Model performance as a function of n with varying confidence

set. Each prediction produces a decision, if d_x was indeed published by the author or not, and a confidence score. We run 2 experiments to study the effect of the threshold τ and n, the number of top similar word embeddings, on the performance of the model. In each experiment, we calculate the precision, recall and F1-score. For each experiment, we also vary the minimum confidence needed to consider a prediction as a valid prediction.

Varying the confidence. Varying the confidence is omitting any prediction with an associated confidence below a certain value. In our case, increasing the minimum confidence required improves the precision on the expense of recall. The results are shown in Figures 2 and 3. When no minimum confidence was required, the model had a precision of 0.53 and an F1-score of 0.26. Increasing the minimum required confidence drastically changes the model's performance when increasing n and τ, reaching a precision of 0.92 while the f1-score suffers due to low recall, the f1-score and recall have a value of 0.02.

Varying τ. τ affects the comparison margin between s_x and sp_x. τ is altered by changing the leniency parameter α. We variate α with the following values: 0.06, 0.09, 0.12, 0.15, 0.18, 0.21, 0.24, 0.27, 0.3. The results of this experiment are shown in Figure 2. We notice that for all confidence values, precision, recall and consequently, f1-score increase as we increase τ. This is because it adds more tolerance for accepting s_x values. However, increasing τ too much would result in a decrease in precision, as more and more s_x values will be accepted.

Varying n. Changing the value of n adds more documents of relevance from the opinion history to compare with d_x. We varied n with the following values: 6, 9, 12, 15, 18, 21, 24, 27, 30. The results of this experiment are shown in Figure 3. We notice that when applying confidence restrictions, the precision increases with the increase of n, on the expense of recall. Precision increases as we increase n and the confidence restriction until it reaches 0.92, for $n = 30$ and confidence = 0.9, while recall suffers drastically, decreasing from 0.52 to 0.02, and consequently, the f1-score drops to 0.02 as well. This is because having additional documents of relevance narrows down the range of accepted sentiments.

6 Discussion

Our model produces results of high accuracy when choosing high minimum confidence, and high top n. This implies that, given enough documents of high relevance to a topic, we indeed do capture the opinion of an author A towards that topic. The implication being, acquiring enough documents centered around said topic to make an accurate decision. However, we do similarity search on word embeddings that take the semantic aspect of keyphrases into account. So the topics of the documents need not to be of an exact match to that of s_x. In addition, the same logic can be applied when facing a new topic never encountered before; the model can find the closest topic, semantically, within a certain level of confidence, and estimate the opinion accordingly. The accuracy of the model could also be improved by using aspect-based sentiment analysis, rather than extracting the sentiment of an entire docu-

tweets per account	TP	TN	FP	FN	unknowns	c@1 score
10	50.0%	18.2%	5.5%	0%	31.8%	0.8388
20	34.9%	7.3%	4.7%	0%	52.7%	0.6458
50	9.1%	9.1%	4.5%	0%	78.2%	0.3223
100	6.8%	6.8%	3.9	0%	79.9%	0.2481
150	4.3%	5.4%	1.1%	1.1%	91.4%	0. 1821
PAN 15 Datasets	43.529%	10.588%	34.118%	2.353%	12.9%	0.6436

Table 1: SPATIUM-L1 performance the short text dataset and the PAN 15 dataset.
TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative

ment. This would assist in handling documents with more than one keyphrase/topic.

7 Conclusions and Future Work

In this work, we presented a novel approach for combating online fraud in social media by IoT botnets through author verification, which is based on sentiment analysis. We also provided a comprehensive comparison against standard approaches. We explained what is lacking in style-based approaches, and experimented against Twitter data. While the performance is lacking, we see promising results. In the future, we plan on investigating the effect of different sentiments per document, and the effect of authors shifting their opinion over time.

References

[1] A. Abbasi, H. Chen, Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace, ACM Trans. Inf. Syst. 26 (2) (2008) 7:1–7:29.

[2] A. Al-Daraiseh, A. Al-Joudi, H. Al-Gahtani, M. Al-Qahtani, Social networks' benefits, privacy, and identity theft: Ksa case study, International Journal of Advanced Computer Science and Applications 5.

[3] H. Allcott, M. Gentzkow, Social media and fake news in the 2016 election, Journal of Economic Perspectives 31 (2) (2017) 211–36.

[4] J. Bevendorff, B. Ghanem, A. Giachanou, M. Kestemont, E. Manjavacas, I. Markov, M. Mayerl, M. Potthast, F. Rangel Pardo, P. Rosso, G. Specht, E. Stamatatos, B. Stein, M. Wiegmann, E. Zangerle, Overview of PAN 2020: Authorship Verification, Celebrity Profiling, Profiling Fake News Spreaders on Twitter, and Style Change Detection, 2020, pp. 372–383.

[5] J. Bevendorff, M. Hagen, B. Stein, M. Potthast, Bias analysis and mitigation in the evaluation of authorship verification, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2019, pp. 6301–6306.

[6] B. T. Boenninghoff, J. Rupp, R. Nickel, D. Kolossa, Deep bayes factor scoring for authorship verification, ArXiv abs/2008.10105.

[7] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguistics 5 (2017) 135–146.

[8] P. Buddha Reddy, T. Murali Mohan, P. Vamsi Krishna Raja, T. Raghunadha Reddy, A novel approach for authorship verification, in: Data Engineering and Communication Technology, Springer Singapore, 2020, pp. 441–448.

[9] D. Castro Castro, Y. Adame Arcia, M. Pelaez Brioso, R. Muñoz Guillena, Authorship verification, average similarity analysis, in: Proceedings of the International Conference Recent Advances in Natural Language Processing, INCOMA Ltd. Shoumen, BULGARIA, 2015, pp. 84–90.

[10] D. De Roure, K. Page, P. Radanliev, M. Van Kleek, Complex coupling in cyber-physical systems and the threats of fake data, 2019, pp. 11 (6 pp.)–11 (6 pp.).

[11] P. Fraga-Lamas, T. M. Fernández-Caramés, Fake news, disinformation, and deepfakes: Leveraging distributed ledger technologies and blockchain to combat digital deception and counterfeit reality, IT Prof. 22 (2) (2020) 53–59.

[12] M. Grootendorst, Keybert: Minimal keyword extraction with bert. (2020).

[13] O. Halvani, L. Graner, R. Regev, Taveer: An interpretable topic-agnostic authorship verification method, in: Proceedings of the 15th International

Conference on Availability, Reliability and Security, Association for Computing Machinery, New York, NY, USA, 2020.

[14] D. I. HOLMES, The Evolution of Stylometry in Humanities Scholarship, Literary and Linguistic Computing 13 (3) (1998) 111–117.

[15] J. Hoorn, S. Frank, W. Kowalczyk, F. van der Ham, Neural network identification of poets using letter sequences, Literary and Linguistic Computing 14 (3) (1999) 311–338.

[16] C. J. Hutto, E. Gilbert, VADER: A parsimonious rule-based model for sentiment analysis of social media text, in: Proceedings of the Eighth ICWSM, June 1-4, 2014, The AAAI Press, 2014.

[17] P. Juola, E. Stamatatos, Overview of the author identification task at pan 2013, CEUR Workshop Proceedings 1179.

[18] M. Kocher, J. Savoy, Unine at CLEF 2015 author identification: Notebook for PAN at CLEF 2015, in: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015, Vol. 1391 of CEUR Workshop Proceedings, CEUR-WS.org, 2015.

[19] M. Koppel, J. Schler, S. Argamon, Authorship attribution in the wild, Lang. Resour. Evaluation 45 (1) (2011) 83–94.

[20] M. Koppel, J. Schler, S. Argamon, Y. Winter, The "fundamental problem" of authorship attribution, English Studies 93 (2012) 284–291.

[21] M. Koppel, J. Schler, K. Zigdon, Determining an author's native language by mining a text for errors, in: Proceedings of the Eleventh ACM SIGKDD 21-24, 2005, ACM, 2005, pp. 624–628.

[22] M. Koppel, Y. Winter, Determining if two documents are written by the same author, J. Assoc. Inf. Sci. Technol. 65 (1) (2014) 178–187.

[23] M. Paquet-Clouston, O. Bilodeau, D. Décary-Hétu, Can we trust social media data?: Social network manipulation by an iot botnet, in: Proceedings of the 8th International Conference on Social Media & Society, 28-30, 2017, ACM, 2017, pp. 15:1–15:9.

[24] A. Peñas, Á. Rodrigo, A simple measure to assess non-response, in: The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24, The Association for Computer Linguistics, 2011, pp. 1415–1424.

[25] F. Peng, D. Schuurmans, S. Wang, Augmenting naive bayes classifiers with statistical language models, Inf. Retr. 7 (3-4) (2004) 317–345.

[26] M. Reznik, Identity theft on social networking sites: Developing issues of internet impersonation, Touro Law Review 29.

[27] A. Rocha, W. J. Scheirer, C. W. Forstall, T. Cavalcante, A. Theophilo, B. Shen, A. Carvalho, E. Stamatatos, Authorship attribution for social media forensics, IEEE Trans. Inf. Forensics Secur. 12 (1) (2017) 5–33.

[28] R. Schwartz, O. Tsur, A. Rappoport, M. Koppel, Authorship attribution of micro-messages, in: Proceedings of the 2013 Conference on EMNLP, ACL, 2013, pp. 1880–1891.

[29] P. Shrestha, S. Sierra, F. A. González, M. Montes-y-Gómez, P. Rosso, T. Solorio, Convolutional neural networks for authorship attribution of short texts, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers, Association for Computational Linguistics, 2017, pp. 669–674.

[30] T. Solorio, S. Pillay, S. Raghavan, M. Montes-y-Gómez, Modality specific meta features for authorship attribution in web forum posts, in: Fifth IJCNLP, 2011, pp. 156–164.

[31] E. Stamatatos, Authorship verification: A review of recent advances, Research in Computing Science 123 (2016) 9–25.

[32] E. Stamatatos, W. Daelemans, B. Verhoeven, P. Juola, A. Lopez-Lopez, M. Potthast, B. Stein, Overview of the author identification task at pan 2015, 2015.

[33] E. Stamatatos, W. Daelemans, B. Verhoeven, M. Potthast, B. Stein, P. Juola, M. Sanchez-Perez, A. Barrón-Cedeño, Overview of the author identification task at pan 2014, CEUR Workshop Proceedings 1180 (2014) 877–897.

[34] M. van Dam, C. Hauff, Large-scale author verification: Temporal and topical influences, in: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, 2014, p. 1039–1042.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30, 2017, pp. 5998–6008.