

Singapore Management University

## Institutional Knowledge at Singapore Management University

---

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

---

9-2020

### DeepStyle: User style embedding for authorship attribution of short texts

Zhiqiang HU

*University of Electronic Science and Technology of China*

Roy Ka-Wei LEE

*University of Saskatchewan*

Lei WANG

*Singapore Management University, lei.wang.2019@phdcs.smu.edu.sg*

Ee-Peng LIM

*Singapore Management University, eplim@smu.edu.sg*

Follow this and additional works at: [https://ink.library.smu.edu.sg/sis\\_research](https://ink.library.smu.edu.sg/sis_research)



Part of the [Databases and Information Systems Commons](#)

---

#### Citation

HU, Zhiqiang; LEE, Roy Ka-Wei; WANG, Lei; and LIM, Ee-Peng. DeepStyle: User style embedding for authorship attribution of short texts. (2020). *Web and Big Data: 4th International Joint Conference, APWeb-WAIM 2020, Tianjin, China, September 18-20: Proceedings*. 12318, 221-229.

Available at: [https://ink.library.smu.edu.sg/sis\\_research/6018](https://ink.library.smu.edu.sg/sis_research/6018)

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email [cherylds@smu.edu.sg](mailto:cherylds@smu.edu.sg).

# DeepStyle: User Style Embedding for Authorship Attribution of Short Texts

Zhiqiang Hu<sup>1</sup>, Roy Ka-Wei Lee<sup>2</sup>(✉), Lei Wang<sup>3</sup>, Ee-peng Lim<sup>3</sup>, and Bo Dai<sup>1</sup>

<sup>1</sup> University of Electronic Science and Technology of China, Chengdu, China  
zhiqianghu@std.uestc.edu.cn, daibo@uestc.edu.cn

<sup>2</sup> University of Saskatchewan, Saskatoon, Canada  
roylee@cs.usask.ca

<sup>3</sup> Singapore Management University, Singapore, Singapore  
lei.wang.2019@phdcs.smu.edu.sg, eplim@smu.edu.sg

**Abstract.** Authorship attribution (AA), which is the task of finding the owner of a given text, is an important and widely studied research topic with many applications. Recent works have shown that deep learning methods could achieve significant accuracy improvement for the AA task. Nevertheless, most of these proposed methods represent user posts using a single type of features (e.g., word bi-grams) and adopt a text classification approach to address the task. Furthermore, these methods offer very limited explainability of the AA results. In this paper, we address these limitations by proposing DeepStyle, a novel embedding-based framework that learns the representations of users' salient writing styles. We conduct extensive experiments on two real-world datasets from Twitter and Weibo. Our experiment results show that DeepStyle outperforms the state-of-the-art baselines on the AA task.

**Keywords:** Authorship attribution · Style embedding · Triplet loss

## 1 Introduction

**Motivation.** Authorship attribution (AA), the task of finding the owner of a given text, plays a vital role in many applications. Particularly in the context of social media, AA is crucial in tackling the growing problem of online falsehood, vicious sock-puppets, and evidence gathering in forensic investigations. Although the AA task has been widely studied and many features have been explored [6, 7, 12–14, 17], the traditional document-based methods did not perform well on the social media posts as they tend to be shorter and less former [6, 15].

To overcome the short-text challenge, several previous AA works have aggregated multiple social media posts of the same user into a single long document before training a classifier to predict the author labels using the long document

**Electronic supplementary material** The online version of this chapter ([https://doi.org/10.1007/978-3-030-60290-1\\_17](https://doi.org/10.1007/978-3-030-60290-1_17)) contains supplementary material, which is available to authorized users.

features, [4, 7]. Such an approach is, however, not effective in handling AA for individual query or a single social media post, which is necessary for some applications, e.g., detecting if two social media posts sharing a rumor originate from the same person. In a recent study, [15] used n-grams of individual Twitter posts to train a Convolutional Neural Network (CNN) model for AA. While CNN and other deep learning models can achieve improved accuracy for the AA task, they still have several limitations. Firstly, the existing models only utilized single input feature types (e.g., bi-gram) to represent the post. Such approaches neglect the rich types of features which had been explored in previous studies to overcome data sparsity in representing the user’s writing styles from short text [13]. Secondly, the deep learning approach usually offers limited explainability on how some factors contribute to good (or bad) AA results as the models themselves are black boxes.

**Research Objectives.** In this paper, we address the limitations in existing methods and propose DeepStyle<sup>1</sup>, an embedding-based framework specifically designed to learn the user style representations using deep learning. The key idea behind DeepStyle is to extract salient features from the users’ social media posts. At the high level, our proposed DeepStyle is divided into two parts. The first part learns a multi-view representation of a user’s posts using deep neural networks over different types of features. Unlike existing deep learning AA methods which use Cross-Entropy loss as objective function, we adopt Triplet loss [3] to learn the post embeddings such that posts that belong to the same user should be close to one another in the post embedding space, while posts from different users will be pushed further away from one another. The second part involves the use of an aggregation function to combine the post embeddings of a single user into the user’s style embedding. The underlying intuition is that the style embeddings of different users should be far apart from one another in the embedding space, thereby distinguishing the users’ writing styles. Finally, the learned user style embeddings are used to perform the AA task.

There are benefits to learning user style embeddings. Firstly, the user style embedding allows us to perform AA in both unsupervised (i.e., finding the most similar user in the embedding space for a given query post) and supervised (i.e., as input to a classifier) settings. Secondly, the visualization of user style embedding enables us to gain an understanding of the feature differences and relationships between user writing styles. Lastly, beyond AA, the user style embedding can also be used for other forensic investigations such as clone [18] or sock-puppet account detection [2] by finding the nearest neighbors in the embedding space, i.e., user accounts who share most similar writing style.

**Contributions.** Our main contributions in this work consist of the following.

- We propose a novel embedding-based framework called DeepStyle, which utilizes multi-view representations of a user’s post and Triplet loss objective functions to learn users’ writing style for AA.

---

<sup>1</sup> Code implementation: [https://gitlab.com/bottle\\_shop/style/deepstyle](https://gitlab.com/bottle_shop/style/deepstyle).

- Using DeepStyle, we manage to learn the latent representation of users’ writing style. To the best of our knowledge, this is the first work that embeds user’s writing style for AA.
- We conduct extensive experiments on two real-world datasets from Twitter and Weibo. The experiment results show that DeepStyle consistently outperforms state-of-the-art methods regardless of the text’s languages.

## 2 Related Works

Authorship attribution (AA) is a widely studied research topic. Traditionally, feature engineering approaches were used to derive textual-related features from long documents such as emails and news articles. Subsequently, these derived features are used to train a classifier to identify an input document’s owner [5, 16]. Commonly used features in AA includes character n-grams [6, 7, 12, 14], lexical features, syntactical features, and document topics [13, 17]. There are also recent studies that explore various deep learning approaches for AA.

In recent years, the AA task has been studied in the social media context. A comprehensive survey [9] provides an overview of AA methods for social media content. The short social media content poses some challenges to the traditional AA methods, [6, 15]. To overcome the short text challenge, the existing methods aggregate multiple social media posts of a user into a single document and before applying the AA methods, [4, 7]. Such an approach cannot adequately handle queries at the single social media post level. Shrestha et al. [15] attempted to overcome this challenge by performing AA using a CNN model with character n-gram features. The n-gram CNN model considers each short text post of a user as input to train a CNN model for AA. In similar studies, Ruder et al. [10] investigated the use of CNN with various types of word and character level features for AA, while Boenningoff et al. [1] proposed a hybrid CNN-LSTM on word level features to perform AA. Although these works achieved reasonably good accuracy for AA in social media, they had utilized single input feature types to represent the users’ posts used in training the deep neural network classifier and offer limited explainability. In this work, we address the limitations of the state-of-the-art by proposing DeepStyle, an embedding-based framework that learns the user style embedding representations using multiple types of features derived from posts to perform AA.

## 3 Proposed Model

### 3.1 Authorship Attribution Problem

Let  $U$  be a set of users in a social media platform,  $U = \{u_1, u_2, \dots, u_N\}$ , where  $N$  is the total number of users in the platform. Each user  $u$  has a set of posts,  $P_u = \{p_{u,1}, p_{u,1}, \dots, p_{u,M_u}\}$ , where  $M_u$  is the number of posts belong to user  $u$ . Given  $U$ ,  $P_u$ , and a query post  $q$ , we aim to predict the most likely author of  $q$ .

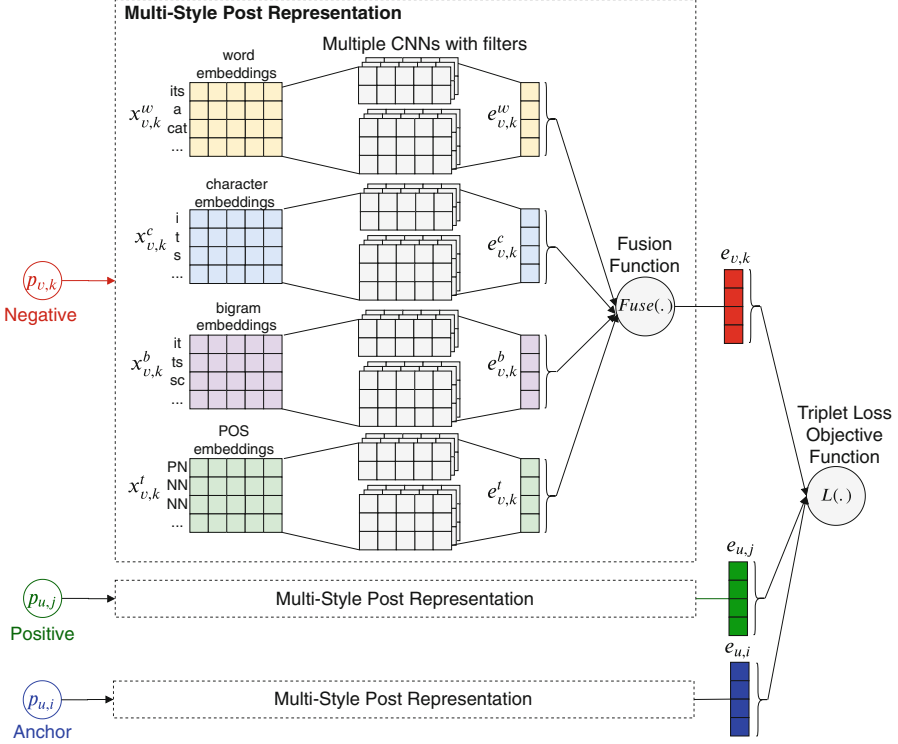


Fig. 1. Post embedding module of the DeepStyle framework

Traditionally, a classifier will be trained using the known posts of all users to predict the most plausible author of  $q$ . In this paper, we adopt a different approach by first learning post embeddings of a user  $u$  with  $M_u$  posts. The symbol  $e_{u,i}$  denotes the post representation of post  $p_{u,i}$ . The learned post embeddings  $E_u = \{e_{u,1}, \dots, e_{u,M_u}\}$  will then be aggregated to form the user style embedding  $s_u$ . Finally, we compute for each user  $u$  the cosine similarity between the query post embedding  $e^q$  and  $s_u$ , denoted by  $\cos(e^q, s_u)$ . Our model predicts the top similar user to be most plausible author of query post  $q$ . In the following, we elaborate both the *post embedding* and *user style aggregation* modules.

### 3.2 Post Embedding Module

Figure 1 illustrates the overall architecture of the post embedding module in our DeepStyle framework. The module learns the post embedding representations using triplet loss. Each triplet is formed by first randomly sampling an anchor post  $p_{u,i}$  and a positive post  $p_{u,j}$  that belongs to the same user  $u$ . We also randomly sample a negative post  $p_{v,k}$ , which is a post belonging to another user  $v$ . With a set of triplets of posts, we learn the post embeddings using *Multi-Style Post Representation*. In this *Multi-Style Post Representation* scheme, we

first perform data pre-processing to extract different types of feature embeddings (e.g., word, character, bigram, POS tags) for each post among the triplets. These feature embeddings will be used as input for a set of Convolution Neural Networks (CNNs) to process and output latent representations of the individual features. Subsequently, a fusion function is used to combine the latent representations and output a post embedding. Finally the three post embeddings,  $e_{u,i}$ ,  $e_{u,j}$ , and  $e_{v,k}$ , will be optimized using the triplet loss objective function.

**Feature Embeddings Extraction.** Previous studies have extensively explored the different types of features, such as words, characters, n-grams, syntactic, and semantic information for AA, [4,13,16]. We leverage on the insights from previous studies to extract the following types of feature embeddings to represent a post  $p_{u,i}$ ,  $p_{u,j}$  or  $p_{v,k}$ . For simplicity, we use  $p_{v,k}$  as an example.

For the four types of input embeddings  $\{x_{v,k}^w, x_{v,k}^c, x_{v,k}^b, x_{v,k}^t\}$ , we obtain four types of output latent representations through multi-level CNN layer, i.e., word latent representation  $e_{v,k}^w$ , character latent representation  $e_{v,k}^c$ , bi-gram latent representation  $e_{v,k}^b$ , and POS tag latent representation  $e_{v,k}^t$ .

**Post Features Fusion Function.** After learning the latent representations of the various types of feature embeddings, we use a fusion function to combine the various latent representations of a post:

$$e_{v,k} = Fuse(e_{v,k}^w, e_{v,k}^c, e_{v,k}^b, e_{v,k}^t) \quad (1)$$

where  $Fuse(\cdot)$  is a typical fusion kernel, and  $e_{v,k}$  has the same dimension as the input latent representations. In our proposed framework, we explore five different implementations of  $Fuse(\cdot)$ : *Mean*, *Max*, *Multilayer perceptron (MLP)*, *Attention* [19], and *Capsule* [11].

### 3.3 User Style Aggregation Module

A user publishes multiple posts on social media platforms. Each post contains some information that defines the user’s writing style, which ultimately would help in authorship attribution. There are multiple ways to combine the post embeddings of a user  $v$  to form the final user style embedding representation,  $S_v$ . In this paper, we introduce three types of aggregators for learning the post embedding of a user: *Mean*, *Max*, and *Capsule* [11].

Using the above three aggregators, we aggregated the users’ post embeddings to form the user style embeddings,  $S = \{s_1, s_2, \dots, s_N\}$ . The user style embeddings will be used to perform AA. We will further evaluate the three aggregation methods in our experiments.

## 4 Experimental Evaluation

### 4.1 Experiment Setup

**Datasets.** We conducted our experiments on two publicly available datasets, namely Weibo<sup>2</sup> and Twitter [14]. The Twitter dataset is widely used in AA

<sup>2</sup> <https://hub.hku.hk/cris/dataset/dataset107483>.

**Table 1.** P@k of various baselines and DeepStyle on Twitter and Weibo datasets. We vary  $k = [1, 10, 50, 100]$ . The improvement of DeepStyle over the best baselines (i.e., CNN-1) is significant at 0.01 level using paired t-test.

Method	Twitter				Weibo			
	P@1	P@10	P@50	P@100	P@1	P@10	P@50	P@100
ToS [13]	9.8	20.4	32.2	43.0	12.1	25.9	38.6	48.3
CNN-1 [10, 15]	18.7	30.7	43.1	49.9	22.0	35.2	48.4	55.9
CNN-2 [15]	20.1	33.7	47.6	55.3	17.5	29.7	41.3	17.5
CNN-W [10, 15]	16.4	27.1	37.6	44.0	21.0	36.5	48.6	54.8
LSTM-1	15.3	25.6	35.9	42.0	19.0	33.6	48.1	55.5
LSTM-2 [15]	15.5	28.0	40.9	48.6	15.8	31.5	43.5	49.8
LSTM-W	11.7	23.7	41.2	48.1	18.7	34.1	48.0	55.2
DeepStyle	<b>21.4</b>	<b>37.1</b>	<b>51.5</b>	<b>59.2</b>	<b>23.9</b>	<b>38.1</b>	<b>51.6</b>	<b>58.9</b>

research [14, 15], while the Weibo dataset, which consists of short Chinese texts, demonstrates the language-agnostic nature of our model. The Twitter dataset contains tweets from 7,026 users, while the Weibo dataset contains posts from 9,819 users. In both datasets, each user has about 1000 posts.

**Baselines.** For evaluation, we compare our proposed DeepStyle model with the following state-of-the-art baselines:

- **ToS:** Based on [13]’s work, ToS method derives content and style-based features from the users’ posts to train a Logistic Regression classifier.
- **CNN:** Following the studies in [10, 15], we train a CNN model each on character features (called CNN-1), bigram features (called CNN-2), and word features (called CNN-W).
- **LSTM:** Long-Short Term Memory (LSTM) model has also been implemented as a baseline in previous study [15] and the model has been successfully used for text classification [8]. We train an LSTM at the character level (LSTM-1), bigram level (LSTM-2), and word (LSTM-W) level.

All the parameters of baselines are empirically set to yield optimal results.

## 4.2 Experiment Results

Table 1 shows the experiment results on Twitter and Weibo datasets. In this experiment, we use 100 posts from each user for training and test on 20 posts from each user. All users on Twitter and Weibo are used in this experiment. Note that the DeepStyle model in this experiment is configured with *Capsule* as the feature fusion function to combine the various latent representations of a post and we select *Mean* as the aggregator to combine the users’ post embeddings into final user style embeddings. The same DeepStyle setting is used for the results reported in Tables 2 and 3. From the results, we observe that DeepStyle

**Table 2.** P@1 for DeepStyle and baselines on varying number of users in Twitter and Weibo.

Method	#Twitter Users					#Weibo Users				
	50	100	500	5000	All	50	100	500	5000	All
ToS [13]	48.3	31.2	22.0	15.7	9.8	50.5	39.5	22.2	16.0	12.1
CNN-1 [10, 15]	60.9	47.1	35.6	23.1	18.7	61.3	52.6	38.8	25.1	22.0
CNN-2 [15]	59.3	46.7	36.5	24.6	20.1	47.6	42.4	30.4	18.7	17.5
CNN-W [10, 15]	50.0	37.1	28.4	17.7	16.4	56.8	48.3	36.2	23.8	21.0
LSTM-1	40.3	29.6	27.3	16.1	15.3	37.5	32.5	30.9	20.1	19.0
LSTM-2 [15]	45.1	34.5	25.2	16.5	15.5	28.1	26.4	19.6	16.8	15.8
LSTM-W	35.0	22.8	19.3	12.0	11.7	35.0	33.4	28.8	19.8	18.7
DeepStyle	<b>64.8</b>	<b>51.2</b>	<b>37.9</b>	<b>25.5</b>	<b>21.4</b>	<b>65.2</b>	<b>56.6</b>	<b>42.8</b>	<b>27.9</b>	<b>23.9</b>

**Table 3.** P@1 for DeepStyle and baselines on varying number of posts for 50 Twitter and Weibo users.

Method	Twitter#posts				Weibo#posts			
	500	200	100	50	500	200	100	50
ToS [13]	47.4	45.1	40.5	36.2	49.2	44.9	40.8	34.7
CNN-1 [10, 15]	59.6	56.0	51.7	49.3	60.5	56.7	52.2	49.1
CNN-2 [15]	61.7	54.1	48.4	44.4	59.4	56.3	51.8	47.5
CNN-W [10, 15]	52.1	48.0	43.3	40.2	58.0	53.1	47.9	44.2
LSTM-1	55.7	49.5	39.8	32.9	54.7	48.7	37.6	32.0
LSTM-2 [15]	53.2	39.4	34.2	30.4	51.0	42.4	35.5	31.4
LSTM-W	47.6	41.5	35.2	30.5	52.8	41.9	36.1	31.2
DeepStyle	<b>62.8</b>	<b>58.7</b>	<b>56.3</b>	<b>50.9</b>	<b>62.5</b>	<b>58.2</b>	<b>56.1</b>	<b>51.1</b>

outperforms the state-of-the-art baselines for different  $k$  values using the  $P@k$  measure. The improvements of DeepStyle over the best baselines (i.e., CNN-1 and CNN-2) are significant at 0.01 level using paired t-test. Similar to the previous study [15], we observe that the CNN models outperform the LSTM models in both datasets. More interestingly, CNN-2 performs better than CNN-1 for the Twitter dataset, while CNN-1 outperforms CNN-2 for the Weibo dataset. This observation could be attributed to the difference in languages; Chinese characters may encode more semantics or definitive user styles. Hence, the Chinese character unigrams can yield better results. DeepStyle, which utilizes multi-style post representations, outperforms all state-of-the-art baselines in both English (i.e., Twitter) and Chinese (i.e., Weibo) datasets.

**Varying Number of Users and Posts.** Similar to previous studies [14, 15], we also want to explore how DeepStyle performs when the AA problem becomes more difficult, i.e., when the number of users increases or when the number of



tweets per author is reduced. Table 2 shows the  $P@1$  of DeepStyle and various baselines varying the number of users in Twitter and Weibo from 50 to all users. For this experiment, we use 100 posts from each user for training and test on 20 posts from each user. We note that although the precision decreases with the increasing number of users, DeepStyle is able to outperform state-of-the-art baselines consistently. Even in the worse case of all users, i.e., over 7,000 and 9,000 Twitter and Weibo users respectively, DeepStyle is still able to outperform the best baselines and the improvements over the best baselines are significant at 0.01 level using paired t-test.

Besides varying the number of users, we also conducted experiments to evaluate the models' performance with different amounts of training data. Table 3 shows the results of the models trained with a different number of posts per user for 50 randomly selected Twitter and Weibo users. We observe the  $P@1$  of various models drops as we reduce the number of posts. However, DeepStyle still consistently outperforms the baselines even with reduced training data.

## 5 Conclusion

In this paper, we proposed a novel embedding-based framework call DeepStyle, which utilized multiple feature types to learn the embedding representation of the user's writing style for authorship attribution (AA). We evaluated DeepStyle using publicly available real-world datasets, and our extensive experiments have shown that DeepStyle is language-agnostic and outperforms the baselines in different experimental settings. For future works, we will like to consider the evolution of the user's writing style by modeling the time-series aspect of the user's posts and consider other better forms of visualizations to explain the user's writing style.

## References

1. Boenninghoff, B., Hessler, S., Kolossa, D., Nickel, R.: Explainable authorship verification in social media via attention-based similarity learning. In: 2019 IEEE International Conference on Big Data (Big Data). IEEE (2019)
2. Bu, Z., Xia, Z., Wang, J.: A sock puppet detection algorithm on virtual spaces. *Knowl.-Based Syst.* **37**, 366–377 (2013)
3. Cheng, D., Gong, Y., Zhou, S., Wang, J., Zheng, N.: Person re-identification by multi-channel parts-based CNN with improved triplet loss function. In: IEEE CVPR (2016)
4. Ding, S.H., Fung, B.C., Iqbal, F., Cheung, W.K.: Learning stylometric representations for authorship analysis. *IEEE Trans. Cybern.* **49**(1), 107–121 (2017)
5. Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. *J. Am. Soc. Inform. Sci. Technol.* **60**(1), 9–26 (2009)
6. Koppel, M., Winter, Y.: Determining if two documents are written by the same author. *J. Assoc. Inform. Sci. Technol.* **65**(1), 178–187 (2014)
7. Layton, R., Watters, P., Dazeley, R.: Authorship attribution for twitter in 140 characters or less. In: IEEE Cybercrime and Trustworthy Computing Workshop (2010)

8. Liu, P., Qiu, X., Huang, X.: Recurrent neural network for text classification with multi-task learning. In: IJCAI (2016)
9. Rocha, A., et al.: Authorship attribution for social media forensics. *IEEE Trans. Inf. Forensics Secur.* **12**(1), 5–33 (2016)
10. Ruder, S., Ghaffari, P., Breslin, J.G.: Character-level and multi-channel convolutional neural networks for large-scale authorship attribution. Insight Centre for Data Analytics. National University of Ireland Galway, Technical Report (2016)
11. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: NIPS (2017)
12. Sapkota, U., Bethard, S., Montes, M., Solorio, T.: Not all character N-grams are created equal: a study in authorship attribution. In: NAACL (2015)
13. Sari, Y., Stevenson, M., Vlachos, A.: Topic or style? Exploring the most useful features for authorship attribution. In: COLING (2018)
14. Schwartz, R., Tsur, O., Rappoport, A., Koppel, M.: Authorship attribution of micro-messages. In: EMNLP (2013)
15. Shrestha, P., Sierra, S., Gonzalez, F., Montes, M., Rosso, P., Solorio, T.: Convolutional neural networks for authorship attribution of short texts. In: EACL (2017)
16. Stamatatos, E.: A survey of modern authorship attribution methods. *J. Am. Soc. Inform. Sci. Technol.* **60**(3), 538–556 (2009)
17. Sundararajan, K., Woodard, D.: What represents "style" in authorship attribution? In: COLING (2018)
18. Xiao, C., Freeman, D.M., Hwa, T.: Detecting clusters of fake accounts in online social networks. In: ACM Workshop on Artificial Intelligence and Security (2015)
19. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: NAACL: HLT (2016)