Automated Twitter Author Clustering with Unsupervised Learning for Social Media Forensics

Sicong Shao
NSF Center for Cloud and Autonomic Computing
The University of Arizona, Tucson, Arizona
sicongshao@email.arizona.edu

Cihan Tunc
NSF Center for Cloud and Autonomic Computing
The University of Arizona, Tucson, Arizona
cihantunc@email.arizona.edu

Amany Al-Shawi
National Center for Cybersecurity Technology
King Abdulaziz City for Science and Technology,
Riyadh, Saudi Arabia
aalshawi@kacst.edu.sa

Salim Hariri
NSF Center for Cloud and Autonomic Computing
The University of Arizona, Tucson, Arizona
hariri@email.arizona.edu

***Abstract*— Twitter is one of the key social media platforms, which is also used for cyber-crimes. Hence, monitoring and detecting the malicious activities of Twitter users is critically important for cybersecurity concerns around the globe since cybercriminals are heavily using Twitter for illegal purpose. It is increasingly common for cybercriminals signing up many accounts while masquerading different users for malicious behaviors. This fact has brought forward the issue of identifying the authors of Twitter accounts. In this paper, we propose a novel approach through a combination of feature extraction methods and then convert high dimensional data to kernel matrix for Twitter author clustering. The experimental results show that our approach can be used to effectively identify the groups among more than one hundred Twitter aliases even without knowing the number of authors.**

Keywords—unsupervised learning; cybersecurity; author identification; author clustering; Twitter; social media

I. INTRODUCTION

The explosive growth of IT infrastructures, cloud systems, mobile devices, and Internet service have resulted in cyber-threats that are growing exponentially in the number and also in the complexity nowadays [19]. Social networking service has been one of the key Internet platforms for rising cyber-threats. Twitter is the most popular microblogging with around 326 million registered users and about 500 million posts per day by 2018 [1]. Twitter allows users to disseminate their opinions, share information, and chat with people across the globe immediately. Although Twitter usage has evident advantages in social media, it also rises cyber-threats and cyber-crimes such as stolen confidential data trade, hacking tools propagation, phishing attacks, and so on [3, 28]. Furthermore, recent studies have found that: organized trolling campaigns turned into a new phase, whereby governments and corporations make effort to influence the opinion surrounding important events (both real and fake) on Twitter [3, 4]. For example, the U.S. Justice Department indicated a Russian state-sponsored media agency named the Internet Research Agency (IRA) to disseminate discord in the US political system [4]. The agency employed hundreds of people to create fake accounts on social media to promote the Russian government's interests in domestic and foreign policy including Ukraine and the Middle East as well as attempting to influence the 2016 US presidential election [11]. Then, Twitter identified 3,814 IRA-linked accounts masqueraded as US citizens to divide voters into a series of issues, such as Black Lives Matter movement, immigration, and feminism [4, 11]. Furthermore, cybercriminals can rely on anonymity methods to ensure the success of cyber-crime [23, 24]. Public hotspots are simple choices of access for malicious activities. Also, one can use the tunneling protocol to tunnel the traffic through VPN service. Sophisticated cybercriminals can even use more advanced techniques like Tor to hide their traces [3]. When these strategies are leveraged by cybercriminals, network forensics may fail [3]. In such a situation, the digital text exposed on social media may be the only clue to identify the attacker(s). Hence, the effective method for monitoring, analysis, and identification has become critically important.

For social media forensics, previous studies mainly focused on identifying authors with a given a set of clear author candidates with digital text samples [6]. However, there are cases when the given author information is unreliable or unavailable. For example, one of the key issues of identifying Twitter users associated with organized trolling campaigns and cyber-crime is that individuals can sign up many Twitter accounts to perform malicious behaviors. Therefore, the authorship information of candidates is not reliable since many candidates may correspond to the same individual. Without reliable training authorship, supervised author identification techniques are hard to be designed for identifying authorship. This fact suggests the requirement for social media forensic techniques that can identify the authors using unlabeled data through unsupervised learning. Author clustering is a promising approach to tackle the author identification problem among unlabeled data, which can be formulated as follows: Given a document collection, our task is to group documents written by the same author. By clustering Twitter user accounts, we can identify different alias accounts posting illegal information actually belong to the same user. Although there exist various author identification studies such as author attribution; the research of author clustering is limited, especially for the Twitter platform. Therefore, we assert that author clustering approach for Twitter is particularly urgent. Hence, in this paper, we propose a novel approach through a combination of different feature extraction methods and, then, convert high dimensional data to kernel matrix to perform clustering.

The remainder of this paper is structured as follows. In Section II we provide background and related work. Section III explains our proposed architecture. The experimental

978-1-7281-5052-9/19/$31.00 ©2019 IEEE

environment and evaluation results are presented in Section IV. Finally, in Section V, the paper is concluded.

II. Background and Related Work

A. Twitter platform

Twitter is a popular microblogging service allowing users to have posts (i.e., tweets) about a broad range of topics. A tweet is a digital text used on Twitter to post opinions, status update, conversation, etc. Follower and following are two account components used by Twitter to structure the social connection between users. The statistic of user's follower and followings are shown on user profile page. Twitter provides a customized timeline of tweets from the given user and users being followed. Furthermore, Twitter also provides several special features in a tweet including hashtags, mentions, replies, and retweets. Hashtags are tags with form of *#topic* for describing arbitrary topic. Retweets build on the authority of other Twitter users, denoting the tweet content appeared on other user's timeline. Mentions and replies are represented using *@username*. Replies is a special form of mentions with inserting *@username* at the beginning of tweet.

B. Author Identification

As social media services have been key platforms of occurrence of cyber-threats, author identification has become an important issue not only in natural language processing (NLP) but also in cybersecurity. Author identification tasks can be divided into author attribution, author verification, author clustering, style change detection, and author diarization. In order to tackle with effectivity issue of author identification, many stylometric approaches which used to capture the writing style of the author have been proposed. Lexical, syntactic, semantic, and content-specific methods have become the four main areas to analyze unique stylometric characteristics of a user [16]. Most author identification works focus on the author attribution problem. For example, Zheng et al. [16] developed an author attribution framework based on writing-style features from lexical, syntactic, word-based, structural and content-specific features. They performed experiments up to 20 of the most active users who frequently posted messages in online newsgroups forums. Abbasi et al. [17] provided an approach called Writeprints based on the extension framework of reference [16]. Instead of using the same features for all authors, they created individual feature sets for each according to the individual's key stylometric features.

Related works of author clustering are still very limited, especially for Twitter platform. Luyckx *et al.* designed an instance of Euclidean distance based centroid clustering to a collection of literary texts [13]. Iqbal *et al.* developed an author clustering method for a collection email to extract the writing style feature from suspects and identify the authorship of anonymous messages [26]. Layton et al. presented an ensemble clustering which can estimate the number of different authors through an iterative positive Silhouette approach [25]. Gómez-Adorno et al. performed a hierarchical clustering for clustering authorship of documents, using features including character n-grams, word n-grams, and stylometric features [12]. For author clustering on Twitter, Yan et al. [8] used character n-grams and function words to extract features of tweets and applied clustering algorithms to identify users. They performed experiments up to 10 users with the best 20.52% accuracy. In summary, there is a lot of room in terms of author identification, specifically author clustering, for the Twitter platform.

III. Author Clustering Architecture

In this section, we describe the architecture of automated unsupervised learning for Twitter author clustering, including automatic Twitter data collection, feature extraction for Twitter author clustering, and unsupervised learning unit. The proposed architecture is shown in Figure 1.

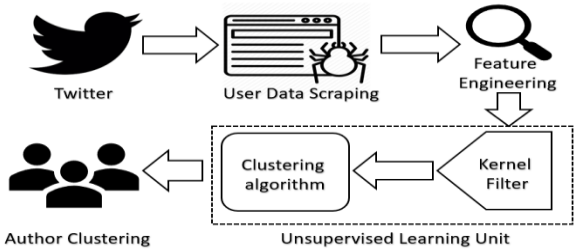

Figure 1. The architecture of automated unsupervised learning for Twitter author clustering

A. Automatic Twitter Data Collection

For Twitter data collection, we leverage the Twint tool that then scrapes given user's tweets, followers, following, etc. [2]. Our autonomic Twitter data collection has the following capabilities: 1) It collects the tweets of the given user as well as the user's followers and followings automatically. 2) It can collect the biography and locations of users if they are provided in the user profile page. 3) It can collect the hashtag and count of likes, count of replies, and count of retweets of each tweet through the option provided by Twint. 4) It can automatically parse the URL in the tweet content through extracting the title of pointed source webpage. With these capabilities, the data collection module can transform the unstructured Twitter data to structured data as a number of different types of CSV files according to the options. In the author identification problem, we use two types CSV files as follows: 1) Username + Tweet content + Date + Time; 2) Username + Bio + Location. The data collection architecture is shown in Figure 2.

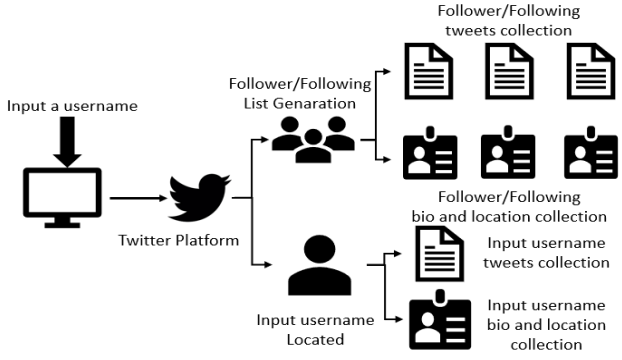

Figure 2. Automated Twitter data collection module

B. Combination of Feature Extraction Methods

For author clustering, we design feature extraction methods based on the following reasons: (1) Even if users attempt to hide and disguise their writing of tweets, they still have certain inherent unconscious writing habits in their messages [16]; (2) The personality characteristics of an individual is relatively stable [21]. Hiding and faking personality features is arduous. Therefore, tweets from users can be used to model their personality characteristics; (3) Malicious users have different levels of cybersecurity knowledge and capability, and different interests in relevant topics. Hence, the used information security terms can be applied to describe the individual's cybersecurity knowledge and attitude; (4) There also exist different public conversation networks in Twitter regarding their replies and mentions messages that can be leveraged for identifying users; and (5) As we perform author clustering in Twitter, activity behaviors can also be used to cluster the aliases; (6) Users have their preference to use emoji to express their emotion. Based on these facts, we have created eleven feature extractors as follows: stop-words, NIST information security terms, letters and numbers and special characters and punctuations, Twitter abbreviations, Twitter emoji usage, activity behaviors, personality insights, replies and mentions, character n-grams, and word n-grams. After feature extraction, we combine the feature sets extracted from all the feature extractors. The architecture of feature extraction modules is presented in Figure 3.

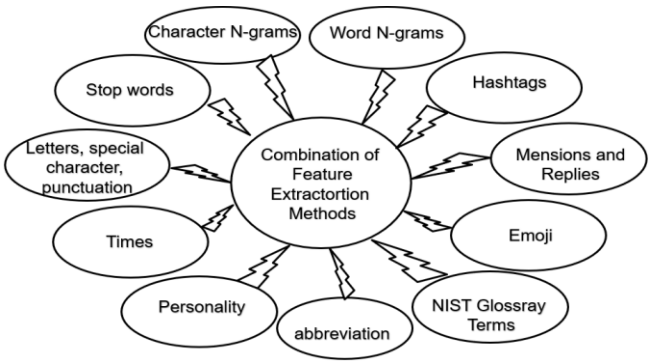

Figure 3. The architecture of feature extraction modules

1) Stop-words

Stop-words have little semantic meaning and are used to express grammatical relationships to construct sentences; thus, they are generally avoided by most search engines due to the high appearance frequency. However, in author clustering to group aliases in the Twitter platform, we hypothesize that stop-words are suitable for distinguishing writing style because stop-words provide information about the user's personal preference of tweet organization. We use the stop-words list provided in [14]. By measuring the frequency of all the stop-words, we can distinguish the characteristic of many Twitter users.

2) NIST Glossary of Information Security Terms

While some Twitter users are cybersecurity novice and have a little cybersecurity knowledge, the others are sophisticated hackers that can utilize advanced cybersecurity techniques. Therefore, the security terms can be used to describe the users' knowledge, their interests, and behaviors. Motivated by helping the users understand information security terminology, NIST has created a repository of key information security terms and definitions, which is regularly updated [10, 15]. We extracted 6,702 terms from the glossary and used the frequency of occurrence of these terms in tweets to describe the user behavior and attitudes from the cybersecurity view.

3) Letters, Numbers, Special Characters, and Punctuations

This feature set is one of the most commonly used features in previous researches [16, 17]. In this research, for each sample, we count the occurrences of 52 letters (case-sensitive), 10 numbers, and 21 special characters (the special characters used in the study of Zheng et al. [16]). Besides, the authors in [16] concluded that the punctuation could provide effective features for authorship analysis, especially for text without editorial normalization. Since tweets are subjected to informal text, the frequency of using punctuation features might also be reliable in Twitter author clustering. We adopted 11 punctuation (".", ",", "?", "!", "'", "\"", ":", ";", "...", "-", "—") in our research. Consequently, we extracted 94 features for each sample through this method.

4) Twitter Abbreviations

We investigated special content features appeared on Twitter in terms of the Twitter common abbreviation. Twitter contains abbreviations like "MT" (modified tweet), "TMB" (tweet me back), "PRT" (please retweet), etc. We used 132 non-repeating abbreviations from the study [18] to represent Twitter user's preference for writing abbreviations.

5) Twitter Emoji

Twitter emojis are used to express emotions. Emojis exist in different genres such as facial expressions, animals, various objects, weather, country, etc. We adopted the most top 200 popular Twitter emoji from the emoji tracker website [20]. Thus, the writing characteristics of users can be analyzed through emoji feature set.

6) Activity Behaviors

There exist multiple factors impacting activity behavior, such as the living time zone and usage habits. Through measuring the number of tweets posted in a certain time interval, a user's activity behavior can contribute to unveil the identity of Twitter user. Moreover, the variance of the number of tweets posted in each time interval can reflect a user's activity degree. Hence, we extract time slot features where a day 48 half-hours are split into 48 equal-size intervals. Then, we count the number of tweets that have been posted in each half-hour interval.

7) Personality Insights

Tweets can be utilized to measure user personality. The personality characteristics are distinguished uniquely for each individual and relatively stable [21]. Using IBM's Personality Insights services [7], [22], we can successfully analyze individual authors' tweet messages and intrinsic personality characteristics to create their personality profiles. Our personality feature extractor can operate using different

languages as IBM Personality Insights service supports multiple languages (e.g., English, Japanese, Korean, Arabic) and in this research we have used English only. By deploying the Personality Insights service of IBM Cloud, we can get an individual user's personality characteristics based on three models: Big Five, Needs, and Values. Big Five model contains the following five primary dimensions: Agreeableness, Conscientiousness, Extraversion, Emotional range, and Openness. Each of these primary dimensions includes six facet features to further distinguish a user. Needs model contains twelve need features, and Values model includes five value features. We selected all the facet features and primary features of Big Five, all the features of Needs model, and all the features of Values model to represent the personality of the Twitter user, which creates 52 features in total. Watson recommends that user provides 1200 words for personality analysis, but providing at least 600 words produces acceptable results. 3000 words are sufficient to achieve the maximum precision [22].

8) Twitter hashtags

Hashtags can be used to simplify content searching on Twitter. Hence, cybercriminals can use hacking words as hashtags. For extracting hashtags features, we use frequency cut-off approach to reduce the high dimensionality of data. Any hashtag appeared at least 10 times is extracted as a hashtag feature.

9) Twitter mentions and replies

While some malicious users are interested in and experienced in discovering vulnerabilities, some focus on diffusing anonymous organization cultures and recruiting new members. Therefore, by analyzing replies and mentions, we can create representation of users' connection and relevance. In order to reduce the high dimensionality of data, we use frequency value cut-off approach. We select the appeared @username feature if appeared at least 10 times.

10) Character n-grams

Character n-grams has been proved to be useful in authorship analysis in emails, newspaper groups, and book writer, etc. [17]. Therefore, we hypothesize that character n-grams can be a useful feature set for clustering Twitter aliases as well. In this study, we used the bag-of-character approach [6] to reduce dimensionality. We extracted the most frequent occurrence of character n-grams features to create features for a text sample. In this study, we have used n $\in$ {2, 3, 4} and the frequency ranked cutoff point is 2,000 for each character n-grams.

11) Word n-grams

The motivation to use word n-grams for author clustering is that Twitter users might have a habit to use some phrases constructed by sequential words. Hence, we hypothesize that word n-grams would be useful to cluster Twitter users. Hence, we use word n-grams features to model Twitter users' habits of using certain word sequences in their tweets. Due to the extremely high dimensionality of word n-gram features, we use the bag-of-words model to represent the feature frequency of word n-grams [27]. A ranked cutoff point is set where 500 most frequent ranked word n-gram (n=2) are selected. Thus, each sample contains 500 most frequent ranked of word 2-gram features.

C. Unsupervised Learning Unit

For successfully clustering Twitter aliases and enhancing the diversity of author clustering model, we use multiple clustering algorithms including k-means clustering, expectation maximization (EM) based Gaussian Mixture Models (GMM), and hierarchical agglomerative clustering.

To improve the performance of clustering models, we use kernel filter to preprocess the dataset. Kernel filter is a preprocessing filter without using label information [9]. Kernel filter converts K dimensions data set into a kernel matrix through applying a kernel function to each pair of samples in the dataset. The new features in the transformed dataset hold the kernel evaluation between a pair of instances, thus generating an $N \times N$ matrix (N is the number of samples). The kernel matrix is given by:

$$\begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & \cdots & K(x_1,x_N) \\ K(x_2,x_1) & K(x_2,x_2) & & \\ \vdots & & \ddots & \vdots \\ K(x_N,x_1) & & \cdots & K(x_N,x_N) \end{bmatrix} \quad (1)$$

where $K(x_i, x_j)$ is kernel function expressed as inner products of two samples. Considering the large numbers of features in our dataset, we use polynomial kernel function of degree 1 which is equivalent to linear kernel. Hence, the dimensionality of the dataset is reduced from K to N.

1) K-Means Clustering

K-Means clustering is an iterative distance-based clustering algorithm. A prespecified number of clusters are required for K-Means clustering. Let $D = \{d_i | i = 1, \ldots, n\}$be a set of samples to be clustered into a set of K-clusters, $C = \{c_k | k = 1, \ldots, k\}$. *K*-Means minimizes the cost function over all *K* clusters as followers:

$$Cost = \sum_{k=1}^{k} \sum_{d_i \in c_k} \|d_i - \mu_k\|^2 \quad (2)$$

where μ_k is the mean of cluster c_k. *K*-Means is a greedy algorithm which converges to a local minimum [30].

2) Hierarchical Agglomerative Clustering

Hierarchical clustering can be either top-down or bottom-up. We use bottom-up in this research. Hierarchical clustering with bottom-up successively agglomerate pairs clusters until all clusters have been merged into a single cluster which includes all samples. Therefore, bottom-up hierarchical clustering is also called hierarchical agglomerative clustering (HAC). When using HAC for flat clustering, we can prespecify the number of clusters *K* and produces *K* clusters.

3) EM based GMM

Gaussian Mixture Models (GMM) is a mixture density models assuming that each component of the probabilistic model is a Gaussian density as followers:

$$p(x|\theta) = \sum_{i=1}^{M} \alpha_i G_i(x|\theta_i) \quad (3)$$

where $x \epsilon \mathbb{R}^d$, $\Theta = (\alpha_1, \cdots, \alpha_{M-1}, \alpha_M; \theta_1, \cdots, \theta_{M-1}, \theta_M)$ satisfy $\sum_{i=1}^{M} \alpha_i = 1, \alpha_i \geq 0$. Gaussian probability density $G_i(x|\theta_i)$ is given as:

$$G_l(x|\theta_l) = \frac{1}{(2\pi)^{d/2}|\sum_l|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu_l)^{\mathrm{T}} \sum_l^{-1} (x-\mu_l)\right\} \quad (4)$$

where $\theta_l = (\mu_l, \sum_l)$. With introducing the latent variable item $Z = \{z_i\}_{i=1}^N$ whose value indicates which component generates the data, GMM assumes the data set $X = \{x_i\}_{i=1}^N$ are generated through M Gaussian components. Hence, $z_i = l$ if a text sample is generated by the l^{th} component. Hence, GMM's parameters can be estimated by the EM algorithm [29]. EM is an iterative procedure algorithm estimating the new parameters in terms of the old parameters. The updating formulas are given as:

$$\alpha_l^{(t)} = \frac{1}{N}\sum_{i=1}^{N} p\left(l|x_i, \Theta^{(t-1)}\right)$$

$$\mu_l^{(t)} = \frac{\sum_{i=1}^{N} x_i p\left(l|x_i, \Theta^{(t-1)}\right)}{\sum_{i=1}^{N} p\left(l|x_i, \Theta^{(t-1)}\right)} \quad (5)$$

$$\sum\nolimits_l^{(t)} = \frac{\sum_{i=1}^{N}\left(x_i - \mu_l{}^{(t)}\right)\left(x_i - \mu_l{}^{(t)}\right)^T p\left(l|x_i, \Theta^{(t-1)}\right)}{\sum_{i=1}^{N} p\left(l|x_i, \Theta^{(t-1)}\right)}$$

where l means the l^{th} component of GMM, and

$$p\left(l|x_i, \Theta^{(t-1)}\right) = \frac{\alpha_l^{(t-1)} G\left(x_i|\theta_l^{(t-1)}\right)}{\sum_{j=1}^{M} \alpha_j^{(t-1)} G\left(x_i|\theta_l^{(t-1)}\right)}, l = 1, \cdots, M \quad (6)$$

$\Theta^{(t-1)} = \left(\alpha_1^{(t-1)}, \cdots, \alpha_{M-1}^{(t-1)}, \alpha_M^{(t-1)}; \theta_1^{(t-1)}, \cdots, \theta_M^{(t-1)}\right)$ and $\Theta^{(t)} = \left(\alpha_1^{(t)}, \cdots, \alpha_{M-1}^{(t)}, \alpha_M^{(t)}; \theta_1^{(t)}, \cdots, \theta_{M-1}^{(t)}, \theta_M^{(t)}\right)$ are parameters of the $(t-1)^{th}$ iteration and $(t)^{th}$ iteration, respectively [29].

4) X-Means Clustering

X-Means is a regularization framework for estimating k, which extending *K*-Means with efficient estimation of the number of clusters [30]. By extending *K*-Means, it uses statistical criteria to make local decisions maximizing the posterior probabilities of model. *X*-Means searches different values of k and scores each model based on Bayesian Information Criterion (BIC). BIC is a model selection method assuming that one of the models is true and it tries to find the model which is most likely to be true in the Bayesian view.

IV. Experiments and Results

To evaluate our method, we performed Twitter author clustering in two cases as clustering Twitter aliases into (1) *known* and (2) *unknown* number of authors. The former case simulates the social media forensic case when law enforcement agency has known the number of authors (who create multiple alias accounts) and alias accounts should be grouped by authorship. The latter case simulates the social media forensic case when the number of authors (who create multiple alias accounts) should be identified and alias accounts should be grouped by authorship.

A. Corpus

For author clustering experiments, a new corpus of potential malicious users was developed using our automatic Twitter data collection module. The steps of the corpus creating are described as follows: (1) The follower's list of three popular Twitter anonymous hacker organization accounts (@OfficialAnonOps, @TheAnonMovement, @AnonymousVideo]) are collected. (2) We included all the followers who use the keywords of "anonymous" or "hack" in their Twitter account biography information. (3) We included all the followers whose total number of tweets is between 1,000 and 7,000. We set 7000 as the maximum number of total tweets because we want to reduce the class imbalance degree of our datasets. (4) We collected all the tweets from followers' timelines except retweet contents since they do not provide information about the users' own writing style and habit. (5) We replaced all the website links and picture links with the keywords of 'URL' and 'PIC', respectively. (6) Next, we only keep the English-speaking followers and remove non-English tweets if they have. After these six steps, the corpus containing total of 638 users is generated. Then, we randomly selected 120 different Twitter user accounts from the corpus and divided them into six datasets for experiments. Each dataset has 20 distinct users as they have different behaviors on Twitter platform. The description of six datasets is shown in Table I. Then we perform non-overlapped segment to each Twitter user's whole tweets to a series of samples which contain the 1000 equal words. Thus, each sample is used to simulate and represent an alias account of a user (we discarded the remaining tweets that are less than 1,000 words). All the samples in each dataset are randomly split into 70% for training set and the remaining 30% for test set.

Table I. DESCRIPTION OF DATASETS

Dataset #	Total # of authors	Total # of samples in dataset	Word count per sample
1	20	586	1,000
2	20	550	1,000
3	20	735	1,000
4	20	546	1,000
5	20	742	1,000
6	20	869	1,000

B. Evaluation Measures

Due to author class labels of samples is available, we use classes-to-clusters evaluation which is a clustering evaluation measures developed by Weka team [9]. The advantage of this clustering evaluation measure is that it treats whether appropriately estimate the number of clusters an important factor. In social media forensic problem, knowing the scale of participants (authors) is critically important, especially for organized cybercrime and organized trolling campaigns. The procedure of classes-to-clusters evaluation is described as follows: after the execution of clustering algorithm is

completed, for evaluating the performance of clustering, the classes-to-clusters evaluation use a brute-force approach to search the minimum error assignment of class labels to clusters with a constraint that one particular class label can only be assigned to one cluster. The clusters without giving class label assignment receive *"No class"*. If there is a situation where the error is equal between the assignment of one particular class to one of several clusters, then the first cluster considered during the search receives the assignment. This mapping is then used to predict class labels for test samples [9]. Note that the label is not used in clustering, while it is used to evaluate the clustering. All the test samples are treated as misclassified if they are clustered to "*No class*". Accuracy, weighted true positive rate and weighted false positive are used to measure the performance of our approach. Weka determines the weighted values by the number of test samples from each class [9].

C. Clustering aliases into known number of authors

We use all feature extractors to extract features that capture different aspects of given tweets. Feature normalization is a method for normalizing the range of features and we apply the rescaling method (min-max normalization) to normalize the range of features in [0, 1]. The formula is given by:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \qquad (7)$$

where x is the original value, x' is the value after normalization, $\min(x)$ is the lower bound value of the range, and $\max(x)$ is the upper bound value of the range.

K-Means, hierarchical agglomerative clustering, and EM based GMM are used with or without kernel filter. For each dataset, same feature extractors with same parameters, and the parameters of the kernel filter and the learning models are same. The results for the comparison of different techniques are shown in Figure 4. We observed that EM based GMM (kernel filter) achieved the best overall performance. We also observed that kernel filter significantly improved the performance of EM based GMM and hierarchical agglomerative clustering. More specifically, EM based GMM (kernel filter) achieved the highest accuracy in four datasets including dataset 1, dataset 2, dataset 3, and dataset 5, with 81.25%, 80.61%, 91.36%, and 80.72% accuracy. EM based GMM (without kernel filter) achieved the best accuracy in two datasets including dataset 4 and dataset 6, with 80.49%, 81.23% accuracy. As the weighted true positive rate, EM based GMM (kernel filter) achieved the best results in five datasets including dataset 1 to dataset 5 with 84.60%, 86.90%, 92.20%, 86.80%, 82.60%, respectively. EM based GMM without kernel filter achieved the highest true positive rate with 91.00% in dataset 6, and also approached the highest true positive rate with 86.80% in dataset 4. For the weighted false positive rate, EM based GMM (kernel filter) achieved best results in dataset 1, dataset 3, dataset 4, dataset 5, and dataset 6 with 0.70%, 0.40%, 0.70%, 0.50%, 0.40%. As the dataset 2, EM based GMM (without kernel filter) achieved the lowest weighted false positive rate of 0.80%. Besides, hierarchical agglomerative clustering (without kernel filter) achieved the worst performance in all dataset.

Figure 4. The author clustering results of six datasets. (a) accuracy. (b) weighted true positive rate. (c) weighted false positive rate.

D. Clustering aliases into self-estimated number of authors

For each experiment of the case of clustering Twitter aliases into self-estimated number of clusters, we use the same training set and test set from the former case. Same feature extractors with same parameters, and same rescaling method are also applied. The parameters of the kernel filter and the learning models in this case are same for each dataset.

EM based GMM and *X*-Means Clustering are considered and each clustering algorithm is performed with or without kernel filter. An important decision for such a case is the model selection for appropriate estimation of the number of authors (clusters) in given data. Two model selection methods, including cross-validation and Bayesian information criterion, are imported for this research. Since we deal with unsupervised learning problem, log-likelihood based cross-validation via EM based GMM and Bayesian Information Criterion (BIC) via *X*-Means are used to estimate the number of authors among many alias accounts.

EM based GMM uses log-likelihood based cross-validation to perform model selection for automatic discovering the number of clusters [9]. The steps to estimate the number of authors in given data is described as follows:

(1) set the initial number of clusters to 1. (2) Training set is randomly split to 10 folds. (3) Performing EM based GMM 10 times using 10 folds. (4) Computing the average log-likelihood over 10 results. (5) The number of clusters is increased by 1 if log-likelihood has increased; otherwise, returning the current number of clusters as the final self-estimated number.

BIC is a model selection method assuming that one of the models is true and it tries to find the model which is most likely to be true in the Bayesian view. For model selection, *X*-Means uses BIC to estimates the number of clusters using training set through penalized likelihood estimation which tries to find a model fitting training data as accurately as possible and also attempts to minimize the complexity of the model [30]. The formula of BIC is given as followers:

$$BIC(C|X) = \mathcal{L}(X|C) - \frac{k(d+1)}{2}\log n \qquad (8)$$

where $\mathcal{L}(X|C)$ is the log-likelihood of the dataset X based on model C, d is the dimensionality and k is the number of clusters [30].

Numbers of clusters detected by different clustering models with/without using kernel filter are presented in Table II. Number of authors (N) and index of datasets are given. K_1, K_2, K_3, K_4 are represented as the number of detected clusters of EM based GMM with kernel filter, EM based GMM without kernel filter, *X*-Means with kernel filter, and *X*-Means without kernel filter, respectively. We observed that kernel filter significantly improved the self-estimated capability of clustering models when estimating the number of clusters (authors) among a large number of aliases in the datasets. More specifically, EM based GMM (kernel filter) accurately estimated the number of clusters in dataset 1, while *X*-Means (kernel filter) accurately determined the number of clusters in dataset 4. As the other datasets, the numbers of detected clusters of EM based GMM (kernel filter) are more close to the number of authors, compared with other methods.

Table II. THE NUMBER OF AUTHORS AND DETECTED CLUSTERS

Dataset #	N	K_1	K_2	K_3	K_4
1	20	20	5	22	4
2	20	13	4	8	2
3	20	16	6	27	2
4	20	19	5	20	3
5	20	15	8	40	4
6	20	23	13	37	9

Figure 5 shows the results of accuracy, true positive rate, and false positive rate of different models. We observed that kernel filter significantly improved the performance of EM based GMM and *X*-Means in terms of all datasets. EM based GMM (kernel filter) approached the best accuracy of 75.57%, 61.21%, 74.39%, 84.30%, 77.78% in dataset 1, dataset 2, dataset 4, dataset 5 and dataset 6. While *X*-Means (kernel filter) approached best 91.36% accuracy in dataset 3, *X*-Means (kernel filter) achieved the highest weighted true positive rate of 92.20%, 75.50%, 89.80%, and 92.90% in dataset 3 to dataset 6. On the other hand, EM based GMM (kernel filter) achieved the highest weighted true positive rate of 78.70% and 61.20% in dataset 1 and dataset 2. For the lowest weighted false positive rate, EM based GMM (kernel filter) achieved 1.10% and 0.90% in dataset 1 and dataset 4 while *X*-Means achieved 2.90%, 0.40%, 0.40% in dataset 2, dataset 3, and dataset 5. According to the results of dataset 6, EM based GMM and *X*-Means both achieved the lowest weighted false positive rate 0.40%.

Figure 5. The author clustering results of six datasets. (a) accuracy. (b) weighted true positive rate. (c) weighted false positive rate.

V. CONCLUSION

The potential anonymity of Internet services, cloud system, and infrastructures can be subverted and used for cybercrime. Authorship analysis is one of the promising approaches to identify attackers who use anonymous communication for threat intent. For example, author clustering can group malicious data generated by the same attacker and therefore provide the capability to assist the Dynamic Data-Driven Application System (DDDAS) for cyber trust analysis [5, 31, 32]. The Twitter platform provides freedom of allowing each user to control many accounts, resulting in an increasing trend that cybercriminals sign up many aliases and fake accounts for malicious behaviors. Hence, it is highly desired to be able to design techniques that

can group alias accounts by authorship. To address this cybersecurity challenge, an automated unsupervised learning approach for Twitter author clustering has been proposed in this paper. We first created our automatic Twitter data collection module to gather potential malicious user data. After that, we perform feature extraction and, then, convert high dimensional data to kernel matrix for Twitter author clustering. Compared to the previous work for Twitter author clustering, our approach can effectively identify the groups among more than one hundred accounts even without knowing the number of authors.

ACKNOWLEDGMENT

This work is partly supported by the Air Force Office of Scientific Research (AFOSR) Dynamic Data-Driven Application Systems (DDDAS) award number FA9550-18-1-0427, National Science Foundation (NSF) research projects NSF-1624668 and NSF-1849113, (NSF) DUE-1303362 (Scholarship-for-Service), National Institute of Standards and Technology (NIST) 70NANB18H263, and Department of Energy/National Nuclear Security Administration under Award Number(s) DE-NA0003946.

REFERENCES

[1] "Twitter by the Numbers: Stats, Demographics & Fun Facts," [Online] URL: https://www.omnicoreagency.com/twitter-statistics/, Accessed: January 2019.

[2] "Github-twintproject" [Online] URL: https://github.com/twintproject/twint

[3] A. Rocha, W. J. Scheirer, C. W. Forstall, T. Cavalcante, A. Theophilo, B. Shen, A. RB Carvalho, and E. Stamatatos, "Authorship attribution for social media forensics," IEEE Transactions on Information Forensics and Security, vol. 12, no. 1, 2017

[4] B. C. Boatwright, D. L. Linvill, and P. L. Warren. "Troll Factories: The Internet Research Agency and State-Sponsored Agenda Building," Resource Centre on Media Freedom in Europe 2018.

[5] S. Hariri, C. Tunc, P. Satam, F. Al-Moualem, and E. Blasch, "DDDAS-Based Resilient Cyber Battle Management Services (D-RCBMS)," In Proceedings of the 2015 IEEE 22nd International Conference on High Performance Computing Workshops (HiPCW), pp. 65-65. IEEE Computer Society, 2015.

[6] E. Stamatatos, "Author identification: Using text sampling to handle the class imbalance problem." Information Processing & Management 44, no. 2, 2008

[7] S. Shao, C. Tunc, A. Al-Shawi, and S. Hariri, "Autonomic Author Identification in Internet Relay Chat (IRC)," In 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), pp. 1-8. IEEE, 2018.

[8] J. Yan, and S. J. Matthews, "Applying clustering algorithms to determine authorship of chinese twitter messages," In IEEE MIT Undergraduate Research Technology Conference (URTC), pp. 1-4. IEEE, 2016.

[9] F. Eibe, M. A. Hall, and I. H. Witten, "The WEKA Workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques," Morgan Kaufmann, 2016.

[10] "Glossary of Key Information Security Terms," [Online] URL: https://csrc.nist.gov/glossary, Accessed: October 2018

[11] Xia, Yiping, Josephine Lukito, Yini Zhang, Chris Wells, Sang Jung Kim, and Chau Tong, "Disinformation, performed: self-presentation of a Russian IRA account on Twitter," Information, Communication & Society (2019): 1-19.

[12] H. Gómez-Adorno, C. Martín-del-Campo-Rodríguez, G. Sidorov, Y. Alemán, D. Vilariño, and D. Pinto. "Hierarchical Clustering Analysis: The Best-Performing Approach at PAN 2017 Author Clustering Task." In International Conference of the Cross-Language Evaluation Forum for European Languages, pp. 216-223. Springer, Cham, 2018.

[13] K. Luyckx, W. Daelemans, and E. Vanhoutte, "Stylogenetics: Clustering-based stylistic analysis of literary corpora," In Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy. 2006.

[14] "Algorithms, 3.5 Searching Applications, stop words" [Online] URL: https://algs4.cs.princeton.edu/35applications/stopwords.txt, Accessed: October 2018

[15] C. Paulsen, "Glossary of Key Information Security Terms. No. NIST Internal or Interagency Report (NISTIR) 7298 Rev. 3 (Draft)," National Institute of Standards and Technology, 2018.

[16] R. Zheng, J. Li, H. Chen, and Z. Huang. "A framework for authorship identification of online messages: Writing - style features and classification techniques," Journal of the American society for information science and technology, vol. 57, no. 3, 2006.

[17] A. Abbasi, and H. Chen, "Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace," ACM Transactions on Information Systems, vol. 26, no. 2, 2008.

[18] "Get Out Your Twittonary: Twitter Abbreviations You Must Know," [Online] URL: https://www.socialmediatoday.com/social-networks/sarah-snow/2015-07-08/get-out-your-twittonary-twitter-abbreviations-you-must-know, Accessed: January 2019

[19] C. Tunc, and S. Hariri. "CLaaS: Cybersecurity Lab as a Service." J. Internet Serv. Inf. Secur. 5, no. 4 (2015): 41-59.

[20] "Emojitracker: realtime emoji use on twitter," [Online] URL: http://emojitracker.com/, Accessed: January 2019

[21] D. A. Cobb-Clark, and S. Schurer, "The stability of big-five personality traits," Economics Letters, vol. 115, no. 1, 2012.

[22] "IBM Watson Personality Insights service," [Online] URL: https://console.bluemix.net/docs/services/personality-insights, Accessed: December 2017

[23] S. Shao, C. Tunc, P. Satam, and S. Hariri, "Real-time irc threat detection framework," In 2017 IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS* W), pp. 318-323. IEEE, 2017.

[24] Z. Fang, X. Zhao, Q. Wei, G. Chen, Y. Zhang, C. Xing, W. Li, and H. Chen, "Exploring key hackers and cybersecurity threats in chinese hacker communities," in Intelligence and Security Informatics (ISI), 2016 IEEE Conference on, 2016, pp. 13-18.

[25] R. Layton, P. Watters, and R. Dazeley, "Automated unsupervised authorship analysis using evidence accumulation clustering," Natural Language Engineering 19, no. 1, 2013.

[26] F. Iqbal, H. Binsalleeh, B. Fung, and M. Debbabi, "Mining writeprints from anonymous e-mails for forensic investigation," digital investigation 7, no. 1-2, 2010.

[27] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, "Class-based n-gram models of natural language," Computational linguistics, vol. 18, no. 4, 1992.

[28] J. Bernard, S. Shao, C. Tunc, H. Kheddouci, and S. Hariri, "Quasi-cliques Analysis for IRC Channel Thread Detection," In International Conference on Complex Networks and their Applications, pp. 578-589. Springer, Cham, 2018.

[29] J. A. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," International Computer Science Institute 4, no. 510, 1998.

[30] D. Pelleg, and A. W. Moore, "X-means: extending k-means with efficient estimation of the number of clusters," In Icml, vol. 1, pp. 727-734. 2000.

[31] E. Blasch, Y. Al-Nashif, and S. Hariri, "Static versus dynamic data information fusion analysis using DDDAS for cyber security trus," Procedia Computer Science 29 (2014): 1299-1313.

[32] E. Blasch, "DDDAS advantages from high-dimensional simulation," In 2018 Winter Simulation Conference (WSC), pp. 1418-1429. IEEE, 2018.