The ISC Int'l Journal of Information Security
ISeCure

July 2019, Volume 11, Number 2 (pp. 173–183)
http://www.isecure-journal.org

Detection of Fake Accounts in Social Networks Based on One Class Classification

Mohammadreza Mohammadrezaei [1], Mohammad Ebrahim Shiri [1,2,*], and Amir Masoud Rahmani [1,3]

[1] *Department of Computer, Borujerd Branch, Islamic Azad University, Borujerd, Iran*
[2] *Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran*
[3] *Computer Engineering Department, Science and Research Branch, Islamic Azad University, Tehran, Iran*

ARTICLE INFO.

Article history:
Received: 2 January 2019
Revised: 31 May 2019
Accepted: 29 July 2019
Published Online: 31 July 2019

Keywords:
Social Networks, Privacy, Fake Accounts, One Class Classification

Abstract

Detection of fake accounts on social networks is a challenging process. The previous methods in identification of fake accounts have not considered the strength of the users' communications, hence reducing their efficiency. In this work, we are going to present a detection method based on the users' similarities considering the network communications of the users. In the first step, similarity measures somethings such as common neighbors, common neighbors graph edges, cosine, and the Jaccard similarity coefficient are calculated based on adjacency matrix of the corresponding graph of the social network. In the next step, in order to reduce the complexity of data, Principal Component Analysis is applied to each computed similarity matrix to provide a set of informative features. then, a set of highly informative eigenvectors are selected using elbow-method. Extracted features are employed to train a One Class Classification (OCC) algorithm. Finally, this trained model is employed to identify fake accounts. As our experimental results indicate the promising performance of the proposed method a detection accuracy and false negative rates are 99.6% and 0%, respectively. We conclude that bringing similarity measures and One Class Classification algorithms into play, rather than the multi-class algorithms, provide better results.

1 Introduction

Nowadays, social networks are highly used and people spend a lot of time on them. Celebrities and big companies utilize networks to communicate with their fans and customers. News agencies also use these networks to broadcast the news [1]. The growth of data transmission and confidential interactions among users might be one of the main reasons why people ignore the negative consequences of sharing personal information on the Internet, especially when information shared as public data for long times [2]. alongside the growing popularity and spread of online social networks, risks and security threats have also increased, consequently this might affect users' privacy and trust [3]. Protecting the privacy of users includes protection of data shared by the users in their profiles, as well as their communications and activities in online social networks. [4, 5].

* Corresponding author.
Email addresses: Mohammadrezaei@iauramhormoz.ac.ir, shiri@aut.ac.ir, rahmani@srbiau.ac.ir
ISSN: 2008-2045

Regarding the vast amount of data exist in social networks, malicious activities and attacks such as phishing, fake accounts creation, and spamming have increased significantly [6]. In the attack of creating a fake account, malicious users introduce themselves as famous people to others [7, 8], and thus they abuse the reputation of individuals or companies, or by creating a fake account they manage to control an account and start to publish false news [9, 10]. Such an attack is mainly intended to obtain personal information from the victim's friends by forging a real profile and increasing trust in friendly environments for further deception of the users in the future [11] detecting and discovering fake accounts through viable approaches can improve security of active users and encourage the producers of social network services to increase the safety level and privacy of their services [12].

Its fifteen years since the first method of detecting fake accounts in social networks was introduced. Since then, many studies have been done and the newer approaches have also challenged some. Although these studies have generally improved network security and its effectiveness, new challenges emerge as network producers are trying to detect fake accounts. Some of these new challenges are as follows.

A one problem might be due to lack of using similarity measures that consider the strength of the mutual friends' network communications among users. In the present study, however, it is believed that the more the mutual friendship network between two users has more connections (more numbers of edges), the greater is the power of users' communication and therefore users' similarity. For instance, we believe that the similarity between v1 and u is higher than that of v2 and u because the number of graph edges of mutual friends between v1 and u is more than the corresponding number for v2 and u. This has not been considered in previous studies. Therefore, the CNGE similarity measure in Section 2.1.2 is defined for this purpose.

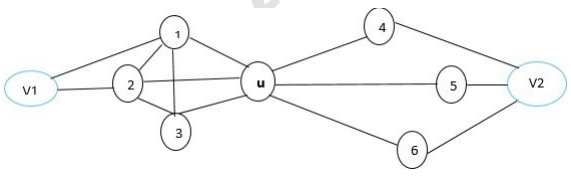

Figure 1. Six neighbors and two strangers of U target users are represented by a node and edges mark their friendship relationships

B. In some of the previous studies, in order to implement suggested methods some problems existed, including: Inaccessibility to dataset of read social networks.

Assuming part of normal users as fake users in order to balance dataset, which is an entirely wrong assumption. To address this problem, the SMOTE algorithm adopted for generation of the artificial samples from the class of the fake users, Therefore, compared to the previous methods, the classification results improved [15].

Novelty of our research consists of the two following parts:

In order to eliminate the challenge of not considering communications strength among users, the present study, based on the theoretical assumptions of the graph, CNGE similarity measure is defined which can appropriately describe this factor.
To solve the data imbalance problem, using One Class Classification algorithms rather than multi-class algorithms are suggested.

The remainder of this paper is organized as follows. In Section 2, basic concepts are provided. Section 3 gives a review of the related literature. The proposed method is elaborated in Section 4. Section 5 shows the experimental results from evaluating the system. Conclusions and some directions for further studies will appear in Section 6.

2 Concepts

2.1 Graph Analysis

Based on the results of the previous studies on similarity measures, this study employed the similarity measures based on common friends or shared connections to form a transition matrix. In what follows, our used measures are described.

2.1.1 Common Neighbors

Let $\Gamma(\nu)$ denotes the neighbors of the v node. Both v and u are more similar if they have more common neighbors [13]. This simple criterion counts the number of common neighbors [14–16].

$$S(v,u) = |\Gamma(v) \cap \Gamma(u)| \tag{1}$$

where $\Gamma(v)$ denotes the set of neighbors of v.

2.1.2 Common Neighbor Graph Edges

Number of edges represents the strength of the relationship between two nodes in the graph [17] . When calculating the similarity of the target user u with user v according to the network, the number of neighbors' edges of u, v is the number of edges of the neighbors of the target user, that is, a graph consisting of all the neighbors of u and all of the edges between the corresponding nodes defined as Equation 2.

$$S(v,u) = \frac{\log|CN(\Gamma(v) \cup \Gamma(u))|}{\log(2|\Gamma(v)|)} \quad (2)$$

Where CN is the number of neighbors they share and $\Gamma(v)$ is the number of edges in the Common neighbor graphs (u, v) and $\Gamma(v)$, respectively.

2.1.3 Jaccard Index

This coefficient is one of the most common metrics in data retrieval and signifies the ratio of the common friends in a union of friends for two nodes[15, 18]. This index is formulated below:

$$S(v,u) = \frac{|\Gamma(v) \cap \Gamma(u)|}{|\Gamma(v) \cup \Gamma(u)|} \quad (3)$$

2.1.4 Cosine Index

The cosine index [15, 19, 20] is defined as

$$S(v,u) = \frac{|\Gamma(v) \cap \Gamma(u)|}{\sqrt{|\Gamma(v) \cdot \Gamma(u)|}} \quad (4)$$

where $|\Gamma(v)|$ denotes the neighbors of v.

2.1.5 L1norm Similarity

This measure is obtained by dividing the overlapping part of the two nodes according to their sizes as shown in Equation (5) [21, 22].

$$S(v,u) = \frac{|\Gamma(v) \cap \Gamma(u)|}{|\Gamma(v) \cdot \Gamma(u)|} \quad (5)$$

2.2 Machine Learning

Machine learning is a procedure of using data for generation of an automatic model in a way that a set of known features is received as input and provides predictions as output. In other words, machine learning regulates and explores methods and algorithms, based on how computers and machines can learn [23]. The main purpose of machine learning is to help a computer (in the most general sense of the word) to gradually find a higher efficiency in conducting its designated task(s) as the amount of data increases [23, 24]. These task(s) might range from automatic fake detection by observation of several samples out of the desired fake to the extraction of the patterns in social networks for users' classification. Machine learning is a rapidly-growing field and is mostly used method in four branches of supervised, unsupervised, reinforcement and semi-supervised learning. Figure 2 illustrates different sub-branches of machine learning are illustrated.

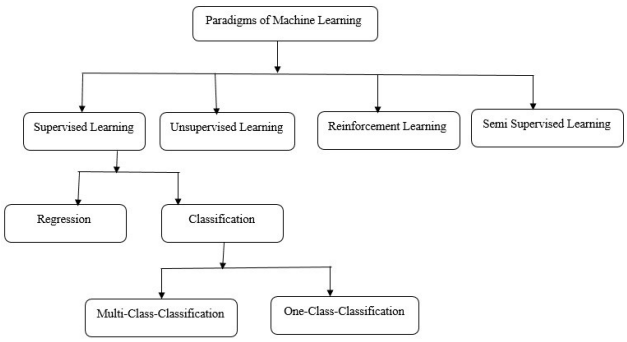

Figure 2. Different sub-branches of the machine learning field

several algorithms used in the proposed method are introduced below.

2.2.1 One Class Classification

One Class Classification (OCC) is a kind of classification in which one of the classes which is considered as a positive a target class is well characterized by samples of educational data[25], whereas the other class which is a negative or a null class. This class does not have any sample but it has a few accessible samples or this is not well defined. This unique situation trains classifiers by defining a positive-order boundary according to the available samples of this class. In recent years, numerous studies have been conducted on OCC, and researchers have proposed several OCC algorithms to deal with various classification problems including unavailability of negative-class samples. In [26, 27], A comprehensive review of such classification techniques is presented. Some algorithms used in the proposed method are introduced below.

A.1. One Class Support Vector Machine

Scholkpf *et al.* [28] first introduced the One Class Support Vector Machine(OSVM) as a useful method for data classification. In this method, using a core function, input data is first mapped to a high-dimensional space and the origin is assumed to be from the class. Then, the margin hyper-planes are frequently found, which best separates the data about the number of observations, density knowledge parameter $\mathcal{P}$, and training parameter l from the origin. v shows the parameters of the support vector fractions and distant points. The formulation of the one-class classifier in principal status v is shown by Equation 6.

$$\min \frac{1}{2}\|W\| + \frac{1}{vl}\sum_{i=1}^{N} E_i - \mathcal{P} \quad (6)$$
$$s,t\,(w, \acute{\varnothing}(x1)) \geq \mathcal{P} - E_i, \quad E_i \geq 0, \;\; 1 \in N \;, \; v \in (0,1)$$

A.2. One Class Algorithm of the Nearest Neighbor

For a test sample of v, let u is the nearest neighbor of the learning data. Also, $d(v,u)$ is the Euclidean distance between v and u, and dy is the maximum distance between the test sample and its nearest neighbor among all the learning data[13]. The output of the nearest neighborhood method can be formulated as follows.

$$r(v) = d(v,u)/dy \tag{7}$$

The value of $r(v)$ when $d(v,u)$ is much smaller than dy can be considered a sign of the anomaly of v. Sample v is placed in the ordinary data or very close to it, and therefore it might show anomalous.

2.3 Principal Component Analysis

Interpretation is one of the main problem in using big data, because high dimension of data makes it difficult to handle them. One technique in order to reduce the size of data is the analysis of the main components. The key challenge in using the Principal Component Analysis technique is to reduce data loss while reducing dimensions. This is done by creation of the independent variables. The variables Which is the most informative the highest variance, and these are new pc variables that lead to a slight reduction in data. For instance, if the x matrix is a $n*p$ matrix containing $x1,\ldots,xp$, the PCA is followed by a linear combination of x columns with maximum variance.

3 Review of Studies on Detecting Fake Accounts

Many studies have already been conducted to detect fake accounts in social networks. These methods are generally divided into three categories as outlined in [6].

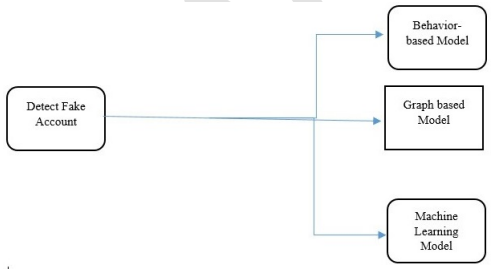

Figure 3. different methods for detection fake accounts

3.1 Behavior-based Methods

It has been prevalent to analyze and evaluate users' behaviors on social networks, and as a result the security risks in the networks increased. Most behavior-based models are derived from clustering algorithms and statistical theories[29]. Wang *et al.* [30] presented an FBI-based social evaluation model. They initially created a user-initiated social impact by examining two main factors of user's own importance and the possibility of affecting others. They designed a social impact adjustment model based on the page rank algorithm by identifying the effect of the friends' influence.

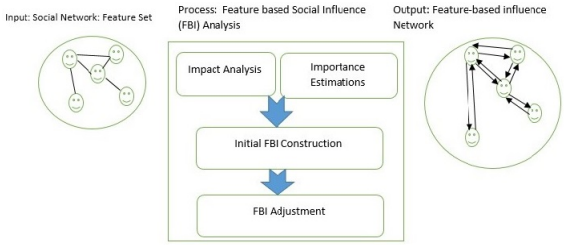

Figure 4. Solution framework [30]

Vigliotti *et al.* [31] proposed a new approach in which behavior is classified as normal or anomaly by checking the p value associated with the occurrence of that behavior.

3.2 Graph-based Methods

Graph is one of the social network analysis methods. In this method, the social network is mapped into a graph, where individuals and organizations form nodes and their communications form the edge[32, 33]. In social networks, this graph is called a social graph. A social network graph can be static or dynamic, labeled or unlabeled[34]. Zhang *et al.* [35] developed a new method for detecting fake accounts. They compared accounts with high shared followers to detect fake accounts. The authors in[36], offered a fake account detection system was offered based on users interactions. The researcher in [37] , Designed a fake account detection system by combining the graph-based and forwarding-based features. Boshmaf *et al.* [38] developed a random walk method to classify fake accounts.

3.3 Machine Learning Methods

In most of the machine learning methods, classifier machine is trained by learning algorithms. Egely *et al.* [39] developed a new approach to detect fake accounts based on users' behavior in the static models trained by Stream-based features. Sangho Lee *et al.* [40] combined the clustering and the classifying methods to provide a new schema for detecting fake accounts.

Kiruthiga *et al.* [41] proposed a detection system for clone-based attacks in social networks, Figure 5 represents the structure of using Bayesian network classifiers, information about the user profile (e.g.,

Basic information and click pattern) and user's active time period in social networks are first classified, then the classes are clustered using the k-means algorithm. These clusters are subsequently sent to the clone spotter in order to diagnose whether the users are fake or normal. The authors also used two similarity measures of cosine and Jaccard to improve their model's efficiency. The proposed structure was implemented on the Facebook social network, where fairly good results were obtained.

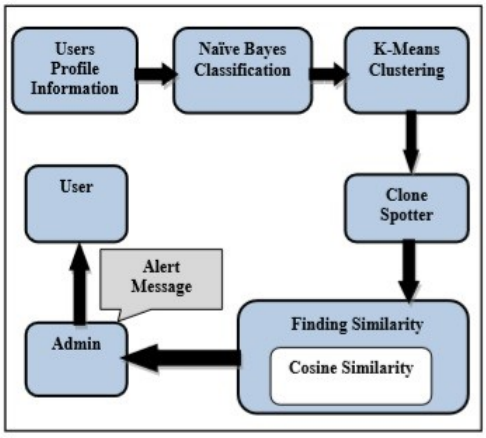

Figure 5. System Architecture for DCA [41]

Cao *et al.* [42] proposed a scalable approach to discover a bunch of fake accounts made by a user. This approach consists of three main steps, represented in Figure 6. These steps are Cluster Builder, Profile Featurizer and Account Scorer.

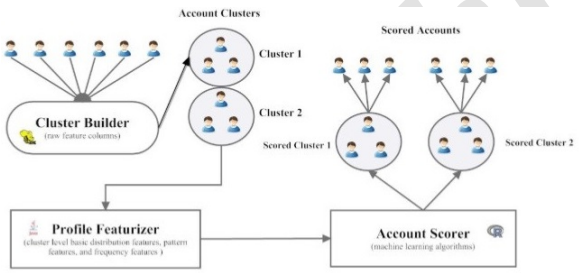

Figure 6. Cao's learning pipeline implementing the fake account clusters detection approach. We assemble accounts into clusters, extract features, train or evaluate the model, and assign scores to the accounts in each cluster [42]

4 Proposed Mehod

This study has introduced a new method to detect fake accounts by combination of the graph-based and machine learning methods. Using the graph, data about user interactions are extracted by defining some measures of similarity. Machine learning is also used to classify the data based on the extracted features. The classification approach used here is based on the one class classification so that the classifier can be well trained according to the features of the normal class members. The flowchart of the proposed method is shown in Figure 7.

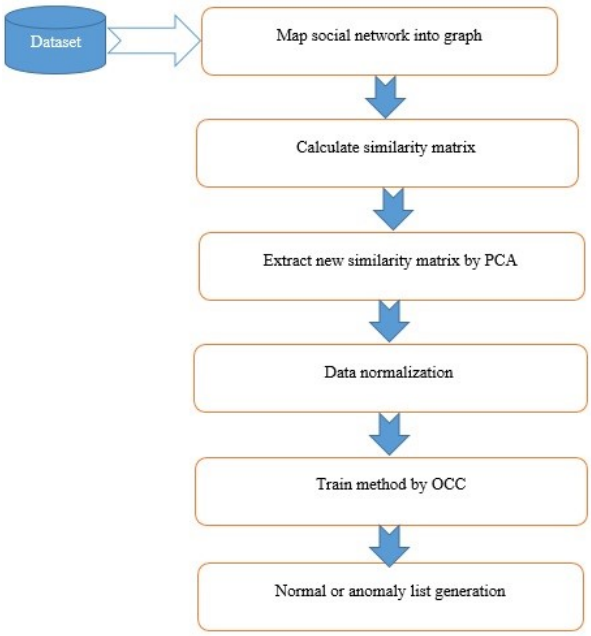

Figure 7. proposed Method

In this section, our proposed two-phased method is described.

(1) Preprocessing phase: It involves mapping of the social network into a social graph, extracting adjacency matrix, calculating similarity features, and extracting matrix of new features.
(2) Training and detection of fake accounts

Phase 1: Data preprocessing involves the following steps.
Step 1: In the first step, the social network is mapped into a graph. In order to map a social network into the graph, a node is created for each user, and an edge is drawn between the two nodes for each connection between the users. Next, for calculation of the adjacency matrix of the graph, if the two users are connected, the value of 1 is designated to the corresponding row and column elements. On the other hand, 0 is assumed for the case where the two users are not connected.
Step 2: Analysis of the previous methods have shown that any features could not distinguish users of a network, alone. Therefore, in the proposed method several features have been used to improve the accuracy of fake accounts detection. The purpose of defining similarity measures is to optimize and to enhance the quality of the extracted features network users. The more an extracted feature contains accurate and dis-

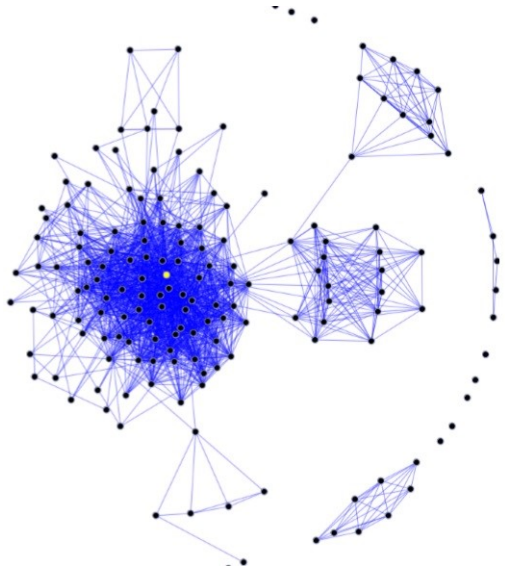

Figure 8. Mapping Social Network into a Graph

tinctive information, the better that feature can be used to detect fake accounts. In this step, for each of the defined measures, such as the number of mutual friends, total number of friends, Common neighbor, Common neighbor graph edges, Jaccard similarity, Cosine similarity, and other measures, the similarity matrix is calculated.

Step 3: In the next step, Principal Component Analysis is used to reduce the complexity of the initial space that works based on the similarity matrices and extraction of the informative features from the similarity matrix that expresses the communication between the users. The columns of the resulting matrices are then sorted in ascending order by their corresponding eigenvalues. A set of highly informative eigenvalues are selected using elbow-method. next, by applying nodes' labels the final dataset is prepared [43].

Phase 2: The following steps shows the phase of training and detecting fake accounts:
Step1: In this step, which is about novelty detection, we first normalize the data using the zero-mean and transforming the unit variance.
Step2: A novel detection method is selected and a machine is trained with the training data. Here, according to the nature of the data, the SVMSch and NN [1] methods are chosen to train the machine, and then the thresholds are calculated.
Step3: The normality or fake of the accounts is determined by entering the new data with respect to the threshold values.

[1] nearest neighbor

5 Simulation of The Results

5.1 Dataset

The Twitter social network is used by hundreds of millions of users. We have also used the Twitter dataset which is available online at: https://github.com/Kagandi/anomalous-vertices-detection/tree/master/data.

5.2 Experiments

In order to simulate our experiments, we have adopted the 2018a MATLAB software, to which ND. Tool `http://www.robots.ox.ac.uk/ $\thicksim $ davidc/publications_NDtool.php` plugin with one-class algorithms is added.

(1) First, we select 1000 nodes from the social network graph of Twitter and extract the associated adjacency matrix. In the adjacency matrix, for every two nodes that are interconnected, the target element is 1, otherwise 0.
(2) We calculate the similarity matrices from the adjacency matrix.
For example, in 1 pseudo code of calculating Cosine similarity matrix mentioned. Here

Pseudocode 1 calculating cosine similarity matrix

```
1: Cos.matrix ← zeros (1000)//create a matrix [1000,1000] with zero string
2: for i = I : 1000 do
3:     Cos.matrix (i,j) ← sqrt(Adj (i,j) * Adj(i,j))
4: end for
5: end
```

Cos.matrix is a cosine similarity matrix between users and *Adj* is the adjacency matrix of the network graph.
(3) Principal Component Analysis is applied to any of the similarity matrices. For example, when the PCA is applied to the cosine similarity matrix, which is a 1000 * 1000 matrix. It gives a new 1000 * 1000 matrix as an output that only the first few columns have information load and are used in the final data; the remainder does not affect the results of the work. In the Discussion Section, the number of columns with the most information load and also the way of choosing them by elbow method[44] has been discussed in details.
(4) The first 5 columns of the values with the highest variance, have been selected. The label for the accounts is applied and the final matrix is prepared.
(5) Using the OCC-based SVMSch and NN algorithms, we train and test the machine. The re-

sults of fake account detection are illustrated below.

This section consists of 5 steps whose pseudo code is illustrated in Algorithm 2.

- Firstly, by using the demo function the data (data. x) and the labels (data. y) are read.
- After loading the data set, data are first divided into three groups of train, validation and test, and then they are normalized.
- Now, we train the learning machine using an algorithm, for example, when ND type = 'SVM-Sch', that is, with a One Class Support Vector Machine, we intend to train.
- Then, we use the trained machine to detect abnormal accounts
- And finally, according to the results of the previous step, we set a threshold value for the distinction of the accounts.

Pseudocode 2 using OCC

$D_{unprocessed} \leftarrow D//D$: dataset
$ND.type \leftarrow' SVMSch'$
$demoND \leftarrow (D_{unprocessed}, ND.type)$
{
$d \leftarrow D_{unprocess}//load(which\,data)$
$d1 \leftarrow d.x \quad //d.x\,all\,dataori$
$d2 \leftarrow d.y \quad //d.y\,class\,labels$
select a $p \in d//a \in d.y\,and\,p \in d.x$
if $a == 0 \quad //regard\,class\,0\,as\,normal$ **then**
    $normal\_class \leftarrow p$
    $else\,fake\_class \leftarrow p$
    }
end if

5.3 Evaluation of Results

In this section, we used actual data of Twitter to evaluate effectiveness of our proposed method. The results of SVMSch and NN algorithms are compared.

The classifiers are evaluated based on the confusion matrix [32]. The variables of the confusion matrix are defined as follows:
True positive (TP): Number of fake nodes that are identified as fake nodes
False Positive (FP): Number of normal nodes that are identified as fake nodes
True Negative (TN): Number of normal nodes that are identified as normal nodes
False Negative (FN): Number of fake nodes that are identified as normal nodes

Table 1. Confusion matrix

		Predicted	
		Fake Accounts	Normal Accounts
Actual	Fake Accounts	TP	FN
	Normal Accounts	FP	TN

The following measures are defined to evaluate the classifier.
Accuracy: the accuracy of a classifier is calculated by dividing the number of the correctly classified objects by the total number of the objects.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{8}$$

Sensitivity: it measures the actual positive values that are correctly detected (for example, the percentage of the fake accounts that are properly detected).

$$\text{sensitivity} = \frac{TP}{TP + FN} \tag{9}$$

False Negative Rate (FNR): FNR or miss rate represents the number of false diagnostic errors.

$$\text{false negative rate} = \frac{FN}{FN + TP} \tag{10}$$

Recall: recall denote the coverage rate of all classified accounts,
AUC: A criterion for evaluating the performance of the classifier and it is equal to the level below the ROC chart. The ROC curve represents the function of classifier accuracy. The ROC curve criteria are shown in equations 14 and 15. It should be noted that if AUC is closer to 1, the performance of classifier will be better.
True negative rate (TNR) =TN/ (TN+FP)
False positive rate (FPR) =FP/ (FP+TN)
True positive rate (TPR) = TP/ (TP+FN)
Experiment 1: First 1000 network nodes were selected and the stages of this method were implemented on them. The results are shown in Table 2 and Figure 9.

Table 2. Comparison of performance of classifier

Algorithm	Accuracy	Sensitivity	FNR
SVMSch	99.6 %	100%	0 %
NN	88.72 %	87.7 %	12.93%

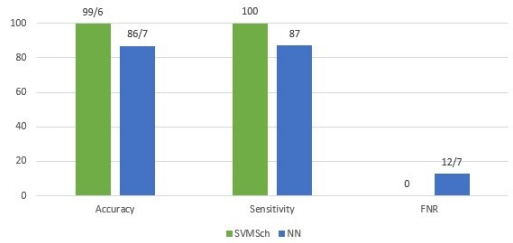

Figure 9. Comparison of performance of classifier

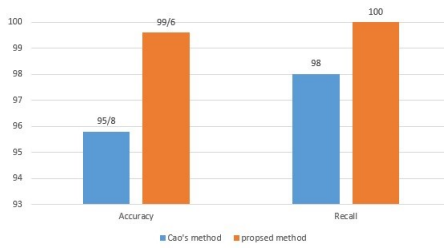

Figure 10. Comparison of proposed method and Cao's method

According to the SVMSch algorithm table, all the fake accounts are properly detected. Besides, the FNR has not assumed any fake accounts to be normal, and therefore its performance is entirely correct. The results of using the nearest neighbor algorithm show that only about 13% of the fake accounts are detected wrongly.

Due to the fact that in the two-class classifiers, unknown samples with no similarity to the data on either of the classes are classified into one of the two classes, the performance of a two-class classifier is inappropriate for these samples and decreases the performance of learning. In order to solve this problem, in this method we use a one class classifier which describes data in a particular domain and defines normal accounts. Each account which is excluded from this description is presumed to be fake. The proposed method outperforms the two-class classifier Table 3.

Table 3. Comparison of One Class SVM (SVMSch) and linear SVM (tow-class SVM) algorithms

Algorithm	Dataset	Accuracy	AUC
SVMSch	1	99.6 %	1
SVM	1	95.8 %	0.98

In Table 4, a comparison is made between the efficiency of our proposed method and that of Cao's method. In Cao's method, support vector machine algorithm [45] has been used to identify a cluster of the fake accounts. The same classifier has also been used in our suggested method, which has proved to be more efficient than Cao's method in detecting fake accounts âĂŞ as shown in Table 3. The higher efficiency of our suggested method can be explained by the use of the one-class algorithms (rather than the multi-class algorithms) which has a better performance in terms of the non-balanced datasets.

Table 4. Comparison of the proposed method and Cao's method using SVM

	Recall	Accuracy	AUC
Proposed method	99 %	99.6 %	1
Cao's method	88 %	89 %	0.898

Table 5. The results of the comparison of SVMSch in two cases, 10/25 columns selected after using PCA

Algorithm	Dataset	No. of columns	Accuracy	Sensitivity	FNR
SVMSch	1	10	99.6 %	100 %	0 %
SVMSch	1	25	95.5 %	99.8 %	0 %

5.4 Discussion

5.4.1 Using Elbow Method

In the fourth step, in order to obtain better results, the elbow method is used to select the number of columns. Based on the results, after applying the PCA algorithm, only 0.0.5 of the data had the highest variance[44, 45] and highest information load. These data were subsequently used for training and testing the classifier.

To prove this, a section of the Twitter dataset with 2000 users was chosen, where the final similarity matrix was a 2000 * 2000 matrix. In PCA, the suggested matrix has 2000 columns.

, the suggested matrix was observed to have 2000 columns. Table 6 clarifies that based on the elbow method, only the first 10 columns had the highest information load. As shown in this table, by selecting the first 10 columns, the classification accuracy was 99.6. However, by increasing the data and selecting 25 columns, the classification accuracy did not change, which indicates that only 0.05% of the data were loaded with information and the use of a larger number of data did not improve the classification results.

5.4.2 Checking Robustness

In order to evaluate the effectiveness of the proposed method in different conditions, different data (the first 1000 nodes, the second 1000 nodes and the fifth 1000 nodes) were trained and classified by the suggested method, and the obtained results were approximately identical.

5.4.3 Expansion Capability

Increasing the number of nodes during the test led to better results. As the number of the nodes in this

Table 6. The results of the comparison of SVMSch in two cases, 10/25 columns selected after using PCA

Algorithm	Dataset	Accuracy	Sensitivity	FNR
SVMSch	1	96.6 %	100 %	0 %
SVMSch	2	99.6 %	100 %	0 %
SVMSch	3	99.4 %	99.8 %	0 %

experiment increased from 1000 to 5000, the predicted results will be improved.

5.4.4 Time Complexity

The time complexity of the proposed method is calculated as follows:

- With regards to the fact that in the proposed method the similarity matrices are extracted from the graph adjacency matrix, that is an n * n matrix in which n is the number of users or sample. The time complexity for extracting each similarity matrix is $O(n^2)$, and according to the calculation of the 5 similarity matrices, in this section the time complexity is $O(n^2)$.
- In the next step, PCA is applied to each similarity matrix and the time complexity of PCA is equal to $O(\min(n^3, P^3))$, where n is the number of samples and p is the number of features and here is $n = p$, so the time complexity is equal to$O(n^3)$ [46].
- In the learning step, the time complexity for OSVM is equal to $O(n^2)$ [47], which at the end the time complexity achieved by the Equation 11 relation.

$$TC = O(n^2) + O(n^3) + O(n^2) = 2O(n^2) + O(n^3) = O(n^3) \quad (11)$$

6 Conclusion

The current paper introduces a new method to detect fake accounts in social networks. In this method, the adjacency matrix was calculated based on the network graph. Moreover, the similarities were derived from the adjacency matrix. By applying principal component analysis and elbow method, new features were extracted. we trained a model and properly predicted fake accounts, Using one-class algorithms. As experimental results of the Twitter dataset show, the accuracy and false negative rates were 99.6% and 0%, respectively.

Finally, we compared the results of aforementioned fake accounts detection in social networks, using one-class classification and multi-class algorithms. we found that one-class classification algorithms had better performance. One of the main limitation of this study is time complexity, although the output of the proposed method is outstanding. Development of efficient implementation approaches to reduce time complexity is suggested for further research.

References

[1] Kagan, D., M. Fire, and Y. Elovici, Unsupervised Anomalous Vertices Detection Utilizing Link Prediction Algorithms. arXiv preprint arXiv:1610.07525, 2016.

[2] Domingo-Ferrer, J., *et al.*, Privacy homomorphisms for social networks with private relationships. 2008. 52(15): p. 3007-3016.

[3] Gao, H., *et al.*, Security issues in online social networks. 2011. 15(4): p. 56-63.

[4] Cutillo, L.A., R. Molva, and T.J.I.C.M. Strufe, Safebook: A privacy-preserving online social network leveraging on real-life trust. 2009. 47(12): p. 94-101.

[5] Van Eecke, P., M.J.C.L. Truyens, and S. Review, Privacy and social networks. 2010. 26(5): p. 535-546.

[6] Adewole, K.S., *et al.*, Malicious accounts: dark of the social networks. Journal of Network and Computer Applications, 2017. 79: p. 41-67.

[7] Krombholz, K., D. Merkl, and E. Weippl, Fake identities in social media: A case study on the sustainability of the facebook business model. Journal of Service Science Research, 2012. 4(2): p. 175-212.

[8] Yu, H., *et al.* Sybilguard: defending against sybil attacks via social networks. in ACM SIGCOMM Computer Communication Review. 2006. ACM.

[9] Subrahmanian, V., *et al.*, The DARPA Twitter bot challenge. 2016.

[10] Van Der Walt, E. and J.J.I.A. Eloff, Using Machine Learning to Detect Fake Identities: Bots vs Humans. 2018. 6: p. 6540-6549.

[11] Fire, M., *et al.*, Online social networks: threats and solutions. 2014. 16(4): p. 2019-2036.

[12] Becker, J.L. and H. Chen, Measuring privacy risk in online social networks. 2009.

[13] Clifton, L.A. and D.S. Yin, Multi-channel novelty detection and classifier combination. 2007: University of Manchester.

[14] Dong, L., *et al.*, The algorithm of link prediction on social network. Mathematical Problems in Engineering, 2013. 2013.

[15] Bank, J. and B.J.W.S.T. Cole, Calculating the jaccard similarity coefficient with map reduce for entity pairs in wikipedia. 2008: p. 1-18.

[16] Mohammadrezaei, M., *et al.*, Identifying Fake Accounts on Social Networks Based on Graph Analysis and Classification Algorithms. 2018. 2018.

[17] Akcora, C.G., B. Carminati, and E. Ferrari, User similarities on social networks. Social Network Analysis and Mining, 2013. 3(3): p. 475-495.

[18] Santisteban, J. and J. Tejada-CÃąrcamo. Uni-

lateral Jaccard Similarity Coefficient. in GSB@ SIGIR. 2015.
[19] Li, Q., *et al.* Mining user similarity based on location history. in Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems. 2008. ACM.
[20] Bayardo, R.J., Y. Ma, and R. Srikant. Scaling up all pairs similarity search. in Proceedings of the 16th international conference on World Wide Web. 2007. ACM.
[21] Gionis, A., P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. in Vldb. 1999.
[22] Akcora, C.G., *et al.*, User similarities on social networks. 2013. 3(3): p. 475-495.
[23] Bishop, C.M.J.P.r. and m. learning, Graphical models. 2006. 4: p. 359-422.
[24] Alpaydin, E., Introduction to machine learning. 2009: MIT press.
[25] Hempstalk, K., E. Frank, and I.H. Witten. One-class classification by combining density and class probability estimation. in Joint European Conference on Machine Learning and Knowledge Discovery in Databases. 2008. Springer.
[26] Khan, S.S. and M.G. Madden. A survey of recent trends in one class classification. in Irish conference on artificial intelligence and cognitive science. 2009. Springer.
[27] Amer, M., M. Goldstein, and S. Abdennadher. Enhancing one-class support vector machines for unsupervised anomaly detection. in Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description. 2013. ACM.
[28] SchÃűlkopf, B., *et al.* Support vector method for novelty detection. in Advances in neural information processing systems. 2000.
[29] Cao, J., *et al.*, Discovering hidden suspicious accounts in online social networks. 2017. 394: p. 123-140.
[30] Wang, G., *et al.*, Fine-grained feature-based social influence evaluation in online social networks. 2014. 25(9): p. 2286-2296.
[31] Vigliotti, M.G. and C.J.S.N. Hankin, Discovery of anomalous behaviour in temporal networks. 2015. 41: p. 18-25.
[32] Al Hasan, M., *et al.* Link prediction using supervised learning. in SDM06: workshop on link analysis, counter-terrorism and security. 2006.
[33] Adewole, K.S., *et al.*, Malicious accounts: dark of the social networks. 2017. 79: p. 41-67.
[34] Savage, D., *et al.*, Anomaly detection in online social networks. 2014. 39: p. 62-70.
[35] Zhang, Y., J.J.S.N.A. Lu, and Mining, Discover millions of fake followers in Weibo. 2016. 6(1): p. 16.
[36] Conti, M., R. Poovendran, and M. Secchiero. Fakebook: Detecting fake profiles in on-line social networks. in Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012). 2012. IEEE Computer Society.
[37] Cao, J., *et al.*, Detection of forwarding-based malicious URLs in online social networks. 2016. 44(1): p. 163-180.
[38] Boshmaf, Y., *et al.*, ÃĮntegro: Leveraging victim prediction for robust fake account detection in large scale OSNs. Computers & Security, 2016. 61: p. 142-168.
[39] Egele, M., *et al.*, Towards detecting compromised accounts on social networks. 2017(1): p. 1-1.
[40] Lee, S. and J.J.C.C. Kim, Early filtering of ephemeral malicious accounts on Twitter. 2014. 54: p. 48-57.
[41] Kiruthiga, S. and A. Kannan. Detecting cloning attack in Social Networks using classification and clustering techniques. in Recent Trends in Information Technology (ICRTIT), 2014 International Conference on. 2014. IEEE.
[42] Cao, Q., *et al.* Aiding the detection of fake accounts in large scale social online services. in Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. 2012. USENIX Association.
[43] Kiselev, V.Y., *et al.*, SC3: consensus clustering of single-cell RNA-seq data. 2017. 14(5): p. 483.
[44] Thorndike, R.L.J.P., Who belongs in the family? 1953. 18(4): p. 267-276.
[45] Pouyan, M.B. and D.J.b. Kostka, Random forest based similarity learning for single cell RNA sequencing data. 2018: p. 258699.
[46] Johnstone, I.M. and A.Y.J.a.p.a. Lu, Sparse principal components analysis. 2009.
[47] Heller, K., *et al.*, One class support vector machines for detecting anomalous windows registry accesses. 2003.

Mohammadreza Mohammadrezaei received his B.S. degree in computer engineering from Islamic Azad University in 2007 and his M.S. degree in software engineering, from science and research branch, IAU, Khuzestan, Iran in 2011. He is a Ph.D. candidate in computer engineering software systems at Islamic Azad University Borujerd Branch, Borujerd, Iran. His research interests include social networks analysis, IOT and data science.

Mohammad Ebrahim Shiri is an assistant professor in the department of computer sciences at Amirkabir University of Technology of Tehran, Iran. He received his Ph.D. from the department of computer sciences at the University of Montreal, Canada in 1999. His current research interests include artificial intelligence, multi-agent systems, intelligent tutoring systems, machine learning, image processing and distributed systems.

Amir Masoud Rahmani received his B.S. in computer engineering from AmirKabir University, Tehran, in 1996, the M.S. in computer engineering from Sharif University of Technology, Tehran, in 1998 and the Ph.D. degree in computer engineering from IAU University, Tehran, in 2005. Currently, he is a professor in the department of computer engineering at the IAU University. He is the author/co-author of more than 200 publications in technical journals and conferences. His research interests are in the areas of distributed systems, Internet of things and evolutionary computing.

The ISC Int'l Journal of Information Security
ISeCure

July 2019, Volume 11, Number 2 (p. 190)
http://www.isecure-journal.org

Persian Abstract

تشخیص حساب‌های کاربری جعلی در شبکه‌های اجتماعی مبتنی بر دسته‌بندی تک کلاسه

محمدابراهیم شیری[۱]، محمدرضا محمدرضایی[۲]، امیرمسعود رحمانی[۳]

[۱]دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی امیرکبیر، تهران، ایران
[۲]دانشکده کامپیوتر، دانشگاه آزاد اسلامی، واحد بروجرد، بروجرد، ایران
[۳]دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد علوم تحقیقات، تهران، ایران

تشخیص حساب‌های کاربری جعلی در شبکه‌های اجتماعی امری چالش برانگیز است. روش‌های پیشین برای کشف حساب‌های کاربری جعلی قدرت ارتباطات میان کاربران را در نظر نگرفته و این باعث کاهش کارایی روش‌های پیشین است. در این تحقیق، ما یک روش تشخیص مبتنی بر شباهت کاربران که ارتباطات شبکه‌ای کاربران را نیز پوشش می‌دهد، ارایه می‌کنیم. در گام اول معیارهای شباهت از قبیل دوستان مشترک، تعداد یال‌های گراف همسایگی، معیار شباهت کسینوس و معیار شباهت جاکارد را از روی ماتریس مجاورت گراف محاسبه می‌شوند. در ادامه به منظور کاهش پیچیدگی‌های محاسباتی و بدست آوردن ویژگی‌های جدید، روش آنالیز مولفه‌های اصلی را روی هر یک از ماتریس‌های شباهت اعمال می‌کنیم. سپس با استفاده از روش البو مجموعه‌ای از مقادیر بردار ویژه که دارای بیشترین بار اطلاعاتی هستند را انتخاب و ماتریس نهایی را تشکیل می‌دهیم. ویژگی‌های استخراج شده برای آموزش الگوریتم دسته‌بند تک کلاسه استفاده می‌شوند. درنهایت این مدل آموزش داده شده برای تشخیص حساب‌های کاربری جعلی استفاده می‌شود. نتایج بدست آمده از آزمایش روش پیشنهادی نشان می‌دهد که دقت تشخیص و نرخ تشخص نادرست به ترتیب ۶.۹۹ % و ۰ % می‌باشند. ما نشان دادیم که تعریف شباهت‌های میان کاربران و همچنین استفاده از الگوریتم‌های تک کلاسه نسبت به چند کلاسه برای آموزش مدل بهتر عمل می‌کنند.

واژه‌های کلیدی: شبکه‌های اجتماعی، حریم خصوصی، حساب‌های کاربری جعلی، دسته‌بندی تک کلاسه.