IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 2, 2020, accepted June 30, 2020, date of publication July 6, 2020, date of current version July 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3007458

An Efficient Victim Prediction for Sybil Detection in Online Social Network

QINGQING ZHOU AND GUO CHEN
College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China

Corresponding author: Guo Chen (guochen@hnu.edu.cn)

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 531107051136, and in part by the National Natural Science Foundation of China under Grant 6187060280.

ABSTRACT With the rapid development of Online Social Networks (OSNs), OSNs have become a rewarding target for attackers. One particularly representative attack is the Sybil attack, Sybil accounts create a lot of malicious activities, which poses a serious threat to the safety of normal users. Many existing Syibl detection mechanisms have preconditions or assumptions, for example, limiting the number of attacking edges. But in general, the assumption is only a handful, often does not hold in real life scenarios. When the assumption is not established, these mechanisms perform poorly. In this work, We propose a scheme that uses victim prediction to improve Sybil detection accuracy. And our solution does not need to be based on any assumptions. First, we designed a victim classifier to predict victims. Then, based on the prediction results, the edge weights in the graph model are modified. Next, trust propagation is performed on the graph model. Finally, sorting all accounts. The experimental results show that our scheme can ensure that the majority of normal users rank higher than Sybils, thus classifying normal users and Sybils.

INDEX TERMS Classification, online social networks, ranking nodes, Sybil detection, victim prediction.

I. INTRODUCTION

In recent years, Online Social Networks (OSNs) have emerged as important platforms for people to communicate across the globe. For example, Twitter, Facebook and Weibo etc. OSNs have become the best sharing and communication platform for people. And OSNs have attracted a large number of users. Facebook is one of the largest social networking sites, Its users reached 1.3 billion, and these users spend approximately 640 million minutes per month on Facebook. Facebook has a variety of users, including celebrities, politicians and ordinary users. However, as the influence and popularity OSNs have increased, A large number of attackers have focused their attention on OSNs. [5], [9], [13], [21], they create a large number of fake identities or hijack a large number of existing legitimate identities, and use them to manufacture various attacks, such as advertising [8], [18], collecting personal privacy information [24], sending spam [7] and so on. The most significant type of intrusions against OSNs is the so-called Sybil attack. Sybil first appeared in distributed networks (such as P2P) to indicate fake identity, and was later introduced into social networks to indicate fake account. In order to create various attacks, Sybil will send a large number of friend requests to normal users. OSNs have become an integral part of user's life, The negative impact of the Sybil attack will not only threaten the security of social networks, but also damage the trust relationship between users. Therefore, the effective detection of Sybil is an urgent problem to be solved. Many schemes have been proposed by industry and academia. We have divided these schemes into two major categories, which are graph-based and machine learning-based [12]. But these solutions have many problems and shortcomings. For example, graph-based solutions generally have prerequisites or assumptions. However, in real scenes, these premises or assumptions may not be satisfied. A lot of research shows that these schemes are easily evaded by attackers in actual scenarios, so their efficiency is very low [4], [22].

In this work, we propose a new victim prediction method to improve Sybil detection accuracy. What user is the victim? On the one hand, the victims are normal users, on the other hand, they accepted Sybil's buddy request. Firstly, we use the bidirectional relationship between users to establish a social network graph model. As the victim accepted Sybil's friend request, there are some connection edges between victims and Sybil accounts in the graph model. These connection

The associate editor coordinating the review of this manuscript and approving it for publication was Kashif Saleem.

edges are attack edges. Secondly, we extracted features from the data set, and used the classifier to predict victims, which can effectively improve the accuracy of account sort, thus improving the efficiency of Sybil detection. Then, based on the prediction results, conducted trust propagation on the graph model, Finally, we sorted nodes according to the node's trust value. We experimentally evaluated the efficiency of our scheme in detecting Sybil using real data. The results show that our scheme has much better detection accuracy than competitors.

In summary, this paper makes the following contributions.

- Our approach extracts six novel features for victims. And these features are in three dimensions, which are user personal information, user behavior and content of message, respectively.
- A novel classifier for victim prediction is proposed, which can greatly improve the efficiency of Sybil detection. We use two-tuples to model the features, resulting in a classifier for predicting victim.

The rest of the paper is organized as follows. Section II discusses related work in OSNs Sybil defense. Section III provides the model overview and graph model. Section IV elaborates on the feature extraction. Section V provides a detailed victim prediction process. Section VI elaborates on the Sybil detection in depth. Section VII presents the experimental results and evaluation. And we conclude in Section VIII with a discussion of our scheme and future work.

II. RELATED WORK

At present, there are many reliable solutions and schemes for defending against Sybil attacks in OSNs have been proposed. We only discuss graph-based schemes, which rely on the attributes of graph to distinguish Sybils and ordinary accounts. Moreover, most of them have the following assumptions.

- The normal region in the graph is fast mixing, as shown in Figure 1. The number of random walks is small, which from the initial state to the stationary state.
- The number of attack edges is limited in the social graph, as depicted in Figure 2. The edges between normal nodes and Sybil nodes are few.

Most of the Sybil detection schemes are based on the social graph techniques. SybilRank [16] sorts users by using

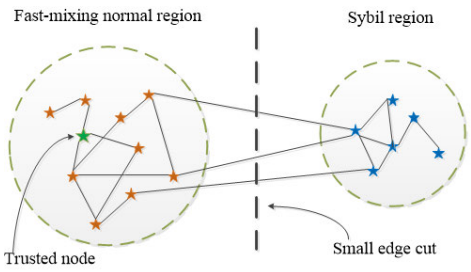

FIGURE 1. The first assumption.

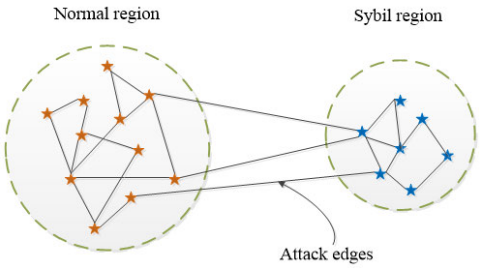

FIGURE 2. The second assumption.

properties of social graph, which based on random walk. Firstly, random walk is power iteration. Distributing the trust value of ordinary nodes to other nodes and normalize. Finally, according to the standardized result, the nodes are ranked. However, this scheme has a time cost of $O(n \log n)$. SybilDefender [20] uses network topology to defend against Sybil. It records the number of times each node is visited during multiple random walks, and then calculates the average and standard deviation to determine whether a node is a normal node. SybilDefender can correctly identify Sybil nodes. However, this scheme only relies on performing a limited number of random walks within the social graph. Reference [10] is another OSN-based protocol. It is intended to address the limitations of SybilGuard through modified random walks. The main concept is to provide a chance to suspect users through an agent walk method to mitigate the problem. However, SybilShield must be based on two assumptions. The first assumption is that the network is fast mixing, and the second assumption is that the number of attack edges is limited. SybilFence [2] analyzes the concept of negative feedback in OSN. It integrates user feedback to limit the social connections of users who receive negative feedback. However, SybilFence has only been tested on synthetic data sets, so there is no way to know the performance in real scenes. Reference [23] proposed Íntegro, which is expanded on the basis of SybilRank. Experimental results show that the detection accuracy of this method reaches 95%. However, this solution needs to be based on assumptions. Reference [6] proposed VoteTrust that applied power iteration to compute the trust probability. It is based on the rationale that a Sybil can be identified by using the friend request acceptance from a normal account. However, VoteTrust does not perform well for some social networks. Such as twitter. Reference [15] proposed SybilRadar. It attempts to improve the accuracy of the early Sybil detection methods, which through introducing a number of stages based on social structural analysis to refine the accuracy of SybilRadar. However, SybilRadar can only be applied to social networks with social structure.

We summarized the above schemes and our own method, as described in Table 1. We can see that most of graph-based schemes are based on the two assumptions mentioned above, that is, the connection between the Sybil area and the normal area is loose. However, due to the complexity of the real scene, these assumptions are not necessarily

TABLE 1. The algorithms and data sets used in each scheme, and whether they are based on assumptions.

Scheme name	Algorithm/methods	Assumption	Datasets
Our own method	Frequency calculation, power iteration	/	Facebook
SybilRank	Random walk/Power iterations	(1,2)	Facebook
SybilDefender	Random walk	(1,2)	Facebook, Orkut
SyiblShield	Modified random walks	(1,2)	MySpace
SybilFence	Feedback mechanism	/	Synthetic dataset
Íntegro	Random forest, power iteration	(1)	Facebook, RenRen, Tuenty
VoteTrust	Power iteration	(1,2)	RenRen
SybilRadar	Modified short random walk	/	Synthetic dataset, Twitter

true [1], [25]. And [3] demonstrated that the existing schemes were inefficient when the assumption was not true. Our solution does not need to be based on any assumptions, so it is more applicable in actual scenarios.

III. MODEL OVERVIEW AND GRAPH MODEL

In this work, we propose a scheme that use victim prediction to improve the accuracy of Sybil detection.

A. OVERVIEW

First, we created a graphical model for each social network. And extracted some features from the user profile and use these features to design a classifier for predicting victims. Then modified the weight of each edge based on the prediction results. Finally, we used improved random walk to spread trust in the social network graph, and sorted the nodes according to the trust value of each node. Our solution can ensure that most Sybil nodes are ranked lower than normal nodes. Thus, Sybil nodes can be separated from the normal nodes. The experimental results also verify the effectiveness of the scheme. The flow chart of the scheme is shown in Figure 3.

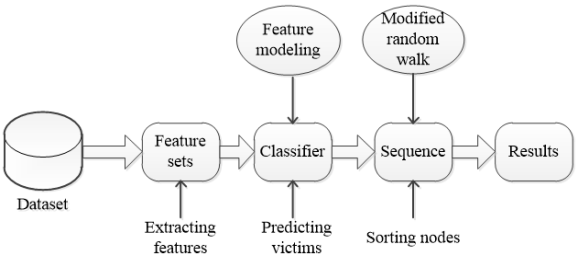

FIGURE 3. The flow chart of our scheme.

B. GRAPH MODEL

As shown in Figure 4, Firstly, we construct an undirected graph $G = (O, E)$ for an OSN, each node $o_i \in O$ indicates a user, and each edge $\{o_i, o_j\} \in E$ indicates a bilateral relationship among o_i and o_j. In the graph G, we use n to denote the number of nodes, that is, $n = |O|$, and m to denote the number of edges, that is, $m = |E|$. We define that each node $o_i \in O$ has a degree $deg(o_i)$, and the degree is equal to the sum of the weights of the edges incident to o_i. Each edge

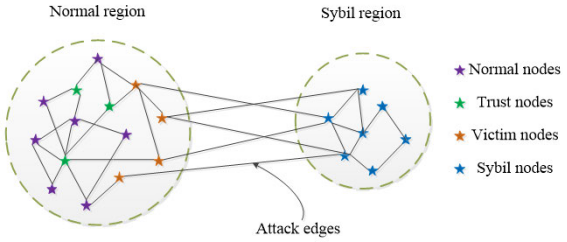

FIGURE 4. The normal region includes normal nodes, victim nodes, trust nodes and their connections. There are Sybil nodes and their connections in Sybil region.

$\{o_i, o_j\} \in E$ has a weight $w(o_i, o_j) \in (0, 1]$, and in the initial case, the weight is set to $w(o_i, o_j) = 1$.

We divide the node set O into two disjoint parts, O_r and O_s, which represent the normal set and the Sybil set, respectively. And we make the subgraph constituted by O_r as the normal region G_r, the normal region G_r contains all normal nodes and the connections between them. Likewise, the another subgraph constituted by O_s as the Sybil region G_s, the Sybil region G_s includes all Sybil nodes and the connections between them. We use E_a to denote the attack edge set, and the attack edge is connected to victims and Sybil nodes, which connects the two regions. Finally, we use O_t to denote the trusted nodes, and we need to assume that the operators of OSNs already know a small trust nodes set, the trust node must be verified as a normal account rather than a victim.

IV. FEATURE EXTRACTION

The victim is a normal user that accepts Sybil friend request. There are some attack edges between victims and Sybils. In general, victims have some common features. We summarized the common features of victims through several experiments. These characteristics cover three dimensions, which are user personal information, user behavior, and message content. As shown in the Table 2, we listed some features and selected 6 features from them to form a feature set for experiments. The six characteristics are the number of friends, user age, the frequency of access, the frequency of update, the number of comments and the URL ratio. Among them, the number of friends and the age of the user are the user's personal information, the frequency of access and

TABLE 2. The relative importance of features.

Features	Description	Type	Importance(%)
Friends	The number of friends	Numeric	100.0
The frequency of updates	The frequency of updates in a period of time	Numeric	90.77
Comments	The number of comments	Numeric	85.9
URL ratio	The ratio of URL contained in message text	Numeric	82.3
User age	The number of days that account registration	Numeric	84.2
The frequency of access	The frequency of access in a period of time	Numeric	80.4
Photos	The number of photos	Numeric	75.1
Gender	The gender of user	2-Categorical	13.8
Messages	The number of messages	Numeric	53.3

the frequency of update are the behavior of the user, and the number of comments and the URL ratio belong to the message content.

And why do we choose these six characteristics? Each feature is contribute to that the normal user become a victim, and we use the relative importance to denote the contribution value. We calculated the relative importance of each feature through several experiments. Eventually, we choose the six features, and their relative importance is greater than other characteristics. Why not use all the features? In fact, we also selected many other features for testing, but found that the combination of these six features has the highest prediction accuracy. Adding other features on top of these six features did not significantly improve the prediction accuracy, so the combination of these six features is the best. Selecting more features is only an increase in computational cost, and does not improve the accuracy.

In contrast to the one-dimensional features, multidimensional features can more effectively improve the accuracy of victim prediction. At the same time, in order to ensure high detection efficiency, we should choose some characteristics that easy access and low-cost. In this paper, the time it takes us to select these six features is $O(1)$.

In this paper, in order to prove that the six features we selected are effective, we analyzed them. We randomly selected 100 victims and 100 non-victims from the real dataset. And the detailed analysis is described below.

- **The number of friends.** In general, attackers prefer to select users with many friends as the target of attack, and in order to become friends with them, the attacker will send a lot of friends request to them. Because these users have more influence than users with fewer friends. We analyzed the number of friends of these 200 users and found that 81 out of 100 victims have more than 100 friends, while only 23 out of 100 non-victims have more than 100 friends. Because the more friends of the victim, the attacker issued a malicious message or activity can be spread to more users. All in all, the users that have more friends are more likely to be victims.
- **User age.** Generally speaking, users with longer registration times are more vulnerable to attacks. Because these users who have registered for a long time have more influence. And We found that 64% of victims registered for more than 5 years, and only 19% of non-victims registered for more than 5 years. Thus, attackers prefer to target users who have registered for a long time.
- **The frequency of access.** Access frequency means the number of times a user is visited by others during a period of time. We analyzed the frequency of access in these 200 users, and found that more than 69% of victims were accessed by friends more than 10 times in a day, while only 22% of non-victims were accessed by friends more than 10 times in a day. The greater the frequency of access, the more active the user is. Attackers are more willing to select these users as targets of attack.
- **The frequency of updates.** The update frequency indicates the number of times the user updates the status within a period of time. In a single day, 71% of victims posted more than 5 posts, while only 34% of non-victims posted more than 5 posts. Thus, attackers are more likely to be friends with these active users. Because, the probability and breadth of active users spreading messages and links are greater than those of inactive users.
- **The number of comments.** We all know that users can write some comments when his friends update his status. And the more comments, the greater the influence of the user. Attackers are very like to be friend with these users. So such users are more likely to be victims. We analyzed the number of comments in these 200 users, and found that victims received significantly more comments than non-victims. In one day, 61% of victims received more than 100 comments, while only 32% of non-victims received more than 100 comments.
- **URL ratio.** In general, users post messages on social networks usually include some URLs, which link to additional resources. And URLs can be linked to various resources, such as videos, pictures and new articles, which can transmit a lot of porn sites, advertising and fishing. Therefore, an attacker has a greater probability to select this type of user as an attack target. Within a day, 83% of posts posted by victims contained URLs, while 21% of posts posted by non-victims contained URLs.

V. VICTIM PREDICTION

Firstly, we need to extract six characteristics from the data set, namely the number of friends, user age, the frequency

of update, the frequency of access, the number of comments and the URL rate. Then model the features to get a classifier, and use the classifier to predict who is the victim. Finally, our classifier is compared with the Naive Bayesian classifier, the decision tree classifier and the random forest classifier. Experimental results show that our classifier performs better than other classifiers, we describe experimental results in the section VII.

We use the $s(o_i)$ to denote the victim score of each account, and judge whether the node is a victim node by the size of victim score $s(o_i)$ and the operation threshold α. If $s(o_i) \geq \alpha$, the node o_i is a potential victim node, and if $s(o_i) < \alpha$, we believe that o_i is a normal node. The whole predicting process is shown in Figure 5.

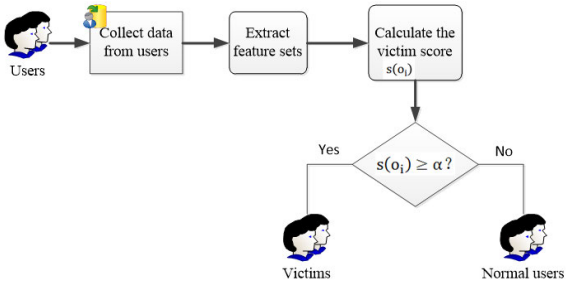

FIGURE 5. The flow chart of victim prediction.

A. MODELING FEATURE

As shown in the Algorithm 1, First, we extract six features from the data set. And we use K to denote the feature model of each user, the elements in K are two-tuple $\langle k_o, v\rangle$, the k_o means a feature, and the v means the value or number of k_o. We use N to denote the total number of users in the dataset. And we define that each feature has a score, the score of each feature is calculated by comparing v and $\overline{K}$, and the calculation of $\overline{K}$ is as follows.

$$\overline{K} = \frac{\Sigma_{i=1}^{\|K\|} v_i}{N}, \tag{1}$$

If $v \geq \overline{K}$, which represents that the feature may be abnormal, then we need to calculate the frequency of the feature, and we define the frequency of each feature as p, which is $p = \frac{v_{k_o}}{N}$. We define the score of each feature as $g(k_o)$, $g(k_o)$ is calculated as follows.

$$g(k_o) = 1 - p = 1 - \frac{v_{k_o}}{N}, \tag{2}$$

And if $v < \overline{K}$, which represents that the feature is normal. Then, we define that the score of the feature is $g(k_o) = 0$. Ultimately, $s(o_i)$ is defined as the average score of all features of each user. and $s(o_i)$ means the victim score of each user. $s(o_i)$ is calculated as follows.

$$s(o_i) = \frac{\Sigma_{j=1}^{f} g_j(k_{o_i})}{f}, \tag{3}$$

And f means the number of species of features of each feature model, and we extracted six types of features. Thus, there are six elements in each feature model K. Therefore, we define $f = 6$ in our scheme.

B. PREDICTING VICTIM

For each user o_i, we calculate it's victim score $s(o_i) \in (0, 1)$ as the probability that it becomes a victim. In addition, we set an operation threshold $\alpha \in (0, 1)$ with an initial value of 0.5 [23]. If $s(o_i) \geq \alpha$, we think o_i is a potential victim, and we say o_i is a normal user if $s(o_i) < \alpha$. Therefore, as long as we extract characteristics of the user o_i, and calculate the victim score of o_i. Then, we can forecast whether o_i a victim is. And we extract n feature sets only need to spend $O(n \log n)$ time.

Algorithm 1 Victim Prediction

Input: Social Graph $G = (O, E)$, the operation threshold α, and the number of species of features f
Output: o_i is a victim or o_i is a normal user
1: $\overline{K} \leftarrow \frac{\Sigma_{i=1}^{\|K\|} v_i}{N}$;
2: if $v \geq \overline{K}$ then
3: $\quad p \leftarrow \frac{v_{k_o}}{N}$
4: $\quad g(k_o) \leftarrow 1 - \frac{v_{k_o}}{N}$
5: else $g(k_o) \leftarrow 0$;
6: $f \leftarrow 6$;
7: $s(o_i) = \frac{\Sigma_{j=1}^{f} g_j(k_{o_i})}{f}$
8: $\alpha \leftarrow 0.5$;
9: if $s(o_i) \geq \alpha$ then
10: $\quad o_i$ is a potential victim;
11: else o_i is a normal user;

VI. SYBIL DETECTION

This section is devoted to show, ranking users and the selection strategy of trusted users.

A. RANKING USERS

In this paper, we designed a classifier to predict victims, which can effectively improve the accuracy of Sybil detection. After obtaining the results of victim prediction, we need to sort nodes. Firstly, we have to select a known trust node as the starting point of the improved random walk algorithm. Then, we define each node has a trust value that is equal to the probability of a random walk landing the node. And we modeled the probability distribution of random walk at each step, which is a process of trust propagation between nodes in graph. In this propagation process, we use the weight $w(o_i, o_j)$ to denote the proportion at which trust may spread from either side of the edge $\{o_i, o_j\} \in E$. And we calculated the trust value for each node by using the power iteration method. And in each iteration, the trust distribution of all nodes need to be calculated. We use $T_n(o_i)$ to represent the trust value of each node $o_i \in O$ after n iterations. And we define the sum of trust value as $T \geq 1$. Initially, we distributed T to all trust

nodes $o_i \in O_t$. This process is shown in Equation (4).

$$T_o(o_i) = \begin{cases} \frac{T}{|O_t|} & if\ o_i \in O_t \\ 0 & otherwise, \end{cases} \tag{4}$$

After initializing the trust value of trust nodes, we have to calculate the trust value $T_n(o_i)$ of node o_i after n iterations. And each node o_i obtains its trust value $T_n(o_i)$ from its all neighbor o_j in the nth iteration. This process is shown in Equation (5).

$$T_n(o_i) = \Sigma_{\{o_i,o_j\}\in E} T_{n-1}(o_j) \cdot \frac{w(o_i, o_j)}{deg(o_j)}, \tag{5}$$

And according to the law of conservation, the node o_i collects the trust value propagated similarly from each neighbor o_j and updates its trust value $T_n(o_i)$. Therefore, the sum of the trust values of all nodes in each iteration is equal. And in this process, T is preserved such that for each iteration $n \geq 1$ we have.

$$\Sigma_{o_i\in O} T_n(o_i) = \Sigma_{o_i\in O} T_{n-1}(o_i) = T, \tag{6}$$

We have to limit the portion of T that escapes the normal region G_r and enters the Sybil region G_s, which can ensure that the trust value of normal users is greater than Syibl. In order to achieve this goal, we have adjusted the propagation rate between nodes, so that the weight of edges of the potential victim nodes is less than that of normal nodes. And the weight of the edge $\{o_i, o_j\} \in E$ is the initial value $w(o_i, o_j) = 1$ if o_i and o_j are not potential victims. If one of o_i and o_j is the victim or both are victims, we need to modify the weight $w(o_i, o_j)$. And the formula of modifying as shown below.

$$w(o_i, o_j) = min\{1, \mu \cdot (1 - max\{s(o_i), s(o_j)\})\}, \tag{7}$$

In the above formula, μ is a scaling parameter, and its initial value is 2. In general, the degrees of each node can affect the trust propagation. When n grows rapidly, the propagation process of trust start to tend to nodes that have high degrees, which indicates that Sybil nodes that have high degrees may get more trust. And trust value of these Sybil nodes may be greater than some normal nodes that have low degrees. Thus, we have to limit this trend. After $k = O(\log n)$ iterations, we use the degree of the node to standardize the trust value. The standardized trust value is the rank value $T'_k(o_i)$ of the node. The calculation of the ranking value of each node is shown below.

$$T'_k(o_i) = \frac{T_k(o_i)}{deg(o_i)}. \tag{8}$$

At last, we rank nodes according to rank values, which is in a descending order. And the experimental results show that most of normal nodes are sorted in the middle and upper part of the sequence, most Sybil nodes are at the bottom of the sequence, thus separating Sybil from normal nodes.

B. THE SELECTION STRATEGY OF TRUSTED USERS

It is very important to seek an efficient selection strategy of trusted nodes, and an efficient selection strategy can improve the accuracy of our scheme. So we choose the Louvain method [19], which can be used to estimate the community structure of the graph model. Then, we select some samples of nodes at random, and these nodes from the communities being detected, we remove potential victims in this process. Furthermore, we inspect sample nodes to verify these nodes are not victims.

VII. SYSTEM MODEL EVALUATION

We used datasets collected from Facebook, and it is more convincing to use real data to analyze and evaluate the scheme. Moreover, our scheme was compared with SybilRank and Íntegro. There are two reasons for why we choose SybilRank and Íntegro. The first reason is that we use a power iteration method similar to SybilRank. but SybilRank constructed an unweighted graph, and we constructed a weighted graph. The second reason is that we and Íntegro both use victim prediction to improve performance. Comparison with SybilRank and Íntegro can highlight the effect of the victim prediction on Sybil detecting. Furthermore, Íntegro is a state-of-the-art Sybil detection approach.

A. DATASETS

The Facebook graph [14] is a connected component sampled via the ''forest fire'' sampling method [11]. And we validated users of the dataset, which can verify whether these users are still using, and we found that 8.2% of users' own cancellation or were banned by the Facebook operators. After we exclude these users, there are 8,586 users in dataset. There are various types' users of Facebook in dataset, including 45.4% of female users and 54.6% of male users, and these users are located in 2,002 cities in 133 countries all over the world.

Finally, we got three graph samples of Facebook. There are two parts contained in each graph, the Sybil region and the normal region. And the first graph includes 65 Sybils with 2,145 connections (the Sybil region), 3,102 normal users with 9,365 connections (the normal area), and 815 attack edges. The second graph includes 64 Sybils with 1,854 connections (the Sybil area), 3,152 normal users with 8,143 connections (the normal region), and 653 attack edges. The third graph includes 2,203 legitimate users with 7,125 connections (the normal region), and this graph does not have Sybil region.

B. CLASSIFIER EVALUATION

In order to get the classifier, we model the extracted features, and then use this classifier to predict victim. The experimental results show that our classifier is more efficient than other classifiers. The classifier was evaluated by TPR (true positive rate), FPR (false positive rate), precision, recall and F-measure. Precision denote the proportion of correctly classified victims, recall denote the coverage proportion of all

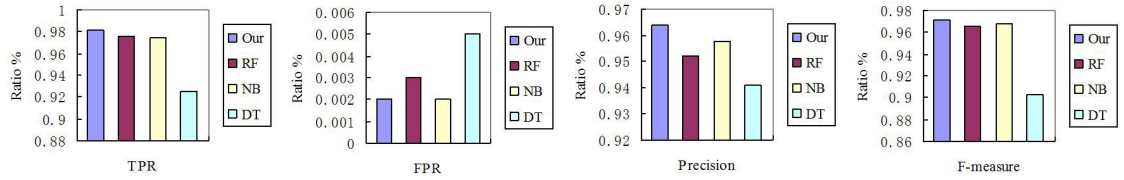

FIGURE 6. The performance of the four classifiers. Our means the our classifier, RF means the Random forest classifier, NB means the Naive Bayesian classifier and DT means the Decision tree classifier.

victims, and F-measure is the weighted harmonic average of precision and recall.

We use these indicators to evaluate the performance of classifiers. We randomly selected 1000 users from the data set. The label rules assigned 97 victims and 903 non-victims. And the experimental environment is built on the server in our laboratory, and the CentOS is installed on the server. This server has a memory capacity of 32 GB and 24 cores. We also imported the Weka package for training classifiers and predicting victims. We chose several classic machine learning methods for comparison, namely random forest, naive Bayes and decision tree. The three classifiers are compared with our classifiers in prediction efficiency. And using the 10-fold cross-validation method to verify the four schemes. The experimental results are listed in Table 3.

TABLE 3. The performance comparison of classifiers.

Classifier	TPR	FPR	Precision	Recall	F-measure
Our classifier	98.2 %	0.2 %	96.4 %	98.2 %	97.1 %
Random forest	97.6 %	0.3 %	95.2 %	97.6 %	96.5 %
Naive Bayesian	97.4 %	0.2 %	95.8 %	97.4 %	96.8 %
Decision tree	92.5 %	0.5 %	94.1 %	92.5 %	90.3 %

All classifiers exhibit high TPR, low FPR and high F-measures, meaning that classifiers perform well and their comprehensive considerations are quite robust. The TPR, FPR, precision, Recall and F-measure of our classifier are 98.2%, 0.2%, 96.4%, 98.2%, and 97.1%, respectively. Experimental results are shown in Figure 6. We can see from the results that the efficiency of our classifier is the best.

C. DETECTION EFFICIENCY EVALUATION

In this section, we compare the performances of our scheme with SybilRank and Íntegro under various attack scenarios. And in the ideal case, Sybils should be ranked lower than normal users. We mainly considered the performance comparison in the two victim attack scenarios. The attackers establish attack edges with targeting users in the first attack scenario, the goal users and attackers have some mutual Sybil friends. We refer to the first scenario as the targeted-victim attack. In the second scenario, regardless of whether the attacker and the target user have common Sybil friends, the attacker randomly establishes an attack edge with the target user. We refer to the second scenario as the random-victim attack.

1) PERFORMANCE METRIC

We chose to use the Receiver Operating Characteristics (ROC) curve to analyze the efficiency of detection scheme. We moved a pivot point from the bottom along the user sorted list. And we think the user is Sybil if it is behind the pivot, otherwise, we think it is a normal user. We measured TPR (true positive rate) and FPR (false positive rate), and use the AUC (the area under its ROC curve) [17] to quantify the probability that any Sybil is sorted lower than any normal user.

2) EVALUATION PROGRAM

In order to evaluate the performance of schemes, we ran SybilRank, Íntegro and our scheme in the two victim attack scenarios, respectively. Firstly, we only set up an attack edge, and then gradually increase attack edge until the maximum number of conditions allowed to repeat the experiment. And we measured the AUC of each scheme at the end of each run. Finally, we calculated the average AUC for each scheme. We selected 200 trusted users that are neither Sybils nor victims. And in the process, we performed $\lceil \log_2(n) \rceil$ iterations, n is the number of nodes.

3) EXPERIMENTAL RESULTS

After several experiments we got the results, and the experimental results show that our scheme is superior to both Íntegro and SybilRank in two scenarios. Especially when the number of attack edges increases, the Íntegro's AUC is a downward trend, but it is always greater than 0.93, and the SybilRank's AUC is significantly decreased. Although our scheme also shows a downward trend, the value of AUC has always been greater than or equal to 0.98.

In each victim attack scenario, the regions are easily separated if normal users are sparsely connect to Sybils. In other words, when the number of attack edges was small, the three schemes all performed well. However, when the number of attack edges gradually increased, the AUC of SybilRank decreased significantly. Íntegro's AUC has also decreased, but it is better than SybilRank. While our model maintains its performance when the number of attack edges increases, the AUC decreases by at most 0.05.

In the first victim attack scenario, with the increase in the number of attack edges, Íntegro's AUC finally remained around 0.96, and the SybilRank's AUC decreases to about 0.71, and our scheme's AUC finally decreases to 0.985. As shown in Figure 7a. It can be seen that the performance

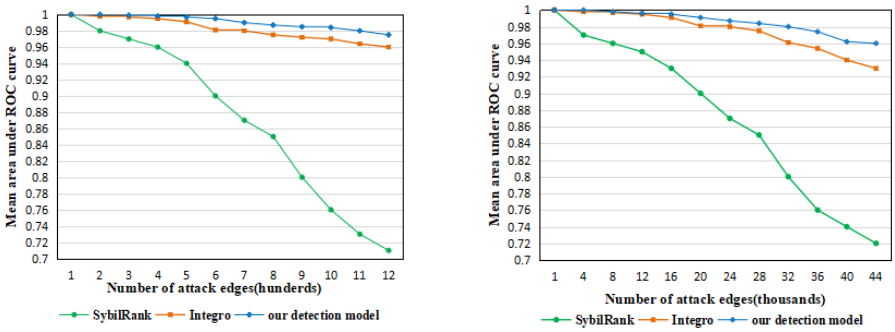

FIGURE 7. The AUC of three schemes in two attack scenarios.

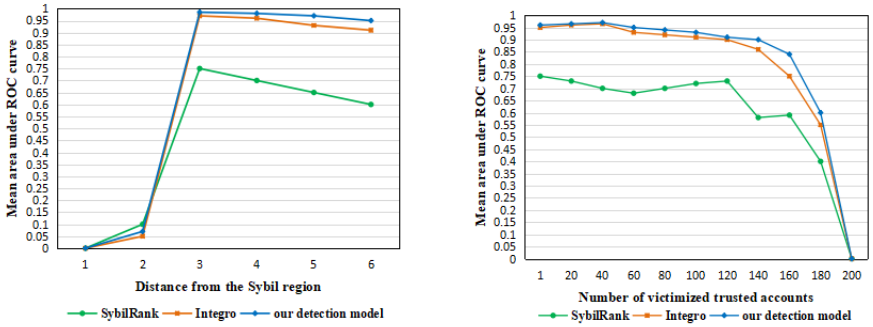

FIGURE 8. The AUC of three schemes in two sensitivity scenarios.

of our scheme is superior to the two existing detection schemes.

And in the second victim attack scenario, the performance of our detection scheme is also superior to the two existing detection schemes. As shown in Figure 7b, as the number of attack edges increases, Íntegro's AUC significantly decreases and approaches 0.93. SybilRank's AUC is significantly reduced, eventually approaching 0.72. But our detection model still maintains a high performance, and its AUC is always greater than 0.98. Thus, based on the experimental results, we can conclude that our detection scheme has good performance under both attack scenarios.

D. SENSITIVITY EVALUATION

We need to consider a scenario where an experienced attacker may know which users the OSN operator trusts. Therefore, attackers can directly establish attack edges with these trusted users, thereby improving Sybil's ranking. We call this scenario a seed-targeting attack.

We also need to consider two attack scenarios. In the first scenario, we use $k+1$ to denote the length of shortest path, which from random trusted nodes to random Sybil nodes, and $k+1$ is also the distance between the Sybil region and the seeds. For instance, if $k=0$, each trusted node is a victim and the distance is 1. We named this attack scenario as a distant-seed attack. And in the second scenario, attackers may don't know which users are trusted by the OSN operators, they randomly select k trusted users as the target. And we named this attack scenario as a random-seed attack.

1) EVALUATION PROGRAM

We simulated the both attack scenario by using the first graph form Facebook, and to evaluate the sensitivity of the three detection schemes. In order to achieve this process, we replaced the endpoint of each attack edge with a normal user, which in the normal region, We randomly selected normal users from candidates. In the first attack scenario, the distance between a candidate user and all trusted users is k nodes. In the second scenario, the candidate user is any trusted user. We performed three detection schemes at different values of k, and measured the AUC of the three detection schemes at each run.

2) EXPERIMENTAL RESULTS

After several experiments we got the results. As shown in Figure 8a, in the first scenario, the three detection models all have a poor performance when the distance is small. However, as the distance increases, our detection model is significantly better than Íntegro and SybilRank. With the increase in the number of victim trusted nodes, in the second attack scenario, the performance of the three detection models has declined. As shown in Figure 8b, it can be seen that our detection model is always better than Íntegro and SybilRank.

In distant-seed attack scenario, the attacker becomes a friend with users, which are at a particular distance from all trusted users. We can know that the three detection models are all sensitive to this attack scenario. And in the random-seed attack scenario, the attacker directly becomes a friend with a subset of trusted users. Thus, we conclude that the three detection models are all sensitive to this attack scenario.

E. RUN TIME ANALYSIS

Usually, for an OSN with n users and m social connection, our scheme need to take $O(n \log n)$ time to complete its calculations. It takes $O(n \log n)$ time to predict victims. Because our scheme training a classifier spends $O(n \log n)$ time, as well as takes $O(n)$ time to calculate the victim score. And the trust spread process of our scheme costs $O(n \log n)$ time, each iteration spends $O(m)$ time, we need to iterate for $O(\log n)$ times. Ultimately, we need to take $O(n \log n)$ time to sort nodes by their rank values. We conclude that the running time of our scheme is $O(n \log n)$, And the running time of Íntegro is $O(n \log n)$, SybilRank has a cost of $O(n \log n)$. The cost of the three schemes is the same.

VIII. CONCLUSION

Nowadays, Sybils in OSNs are experiencing an explosive growth, which seriously compromises the security of users in OSNs. And the crafty operators of Sybil mimic normal users' behavior to hide himself, which makes Sybil's detection more challenging. In order to solve this problem, We proposed an efficient Sybil detection scheme that uses victim prediction to improve the efficiency of Sybil detection. Firstly, we model features to obtain a classifier, which used to forecast victim users. Then, we applied the prediction results to rank users. Finally, according to the ranking results, Sybils can be separated from normal users. And the proposed scheme can help operators of OSNs to effectively detect Sybils.

And as compared to Íntegro and SybilRank, the performance of our detection model is obviously superior to the other two schemes in the both attack scenarios. In addition, Íntegro and SybilRank are currently the best schemes available. However, the running time of our scheme is $O(n \log n)$, and the running time of Íntegro is $O(n \log n)$, SybilRank also has a cost of $O(n \log n)$. The cost of the three programs is the same. Thus, we need to consider how to reduce the time complexity of our scheme in the future work.

REFERENCES

[1] A. Mohaisen, A. Yun, and Y. Kim, "Measuring the mixing time of social graphs," in *Proc. 10th Annu. Conf. Internet Meas. (IMC)*, 2010, pp. 383–389.
[2] Q. Cao and X. Yang, "SybilFence: Improving social-graph-based sybil defenses with user negative feedback," *CoRR*, vol. abs/1304.3819, Apr. 2013. [Online]. Available: http://arxiv.org/abs/1304.3819
[3] D. Koll, M. Schwarzmaier, J. Li, X.-Y. Li, and X. Fu, "Thank you for being a friend: An attacker view on online-social-network-based Sybil defenses," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. Workshops (ICDCSW)*, Jun. 2017, pp. 157–162.
[4] J. R. Douceur, "The Sybil attack," in *Proc. 1st Int. Workshop Peer-Peer Syst.* New York, NY, USA: Springer-Verlag, 2002, pp. 251–260.
[5] H. Gao, J. Hu, T. Huang, J. Wang, and Y. Chen, "Security issues in online social networks," *IEEE Internet Comput.*, vol. 15, no. 4, pp. 56–63, Jul. 2011.
[6] J. Xue, Z. Yang, X. Yang, X. Wang, L. Chen, and Y. Dai, "VoteTrust: Leveraging friend invitation graph to defend against social network Sybils," in *Proc. IEEE INFOCOM*, vol. 12, Apr. 2013, pp. 2400–2408.
[7] K. Thomas, C. Grier, D. Song, and V. Paxson, "Suspended accounts in retrospect: An analysis of Twitter spam," in *Proc. ACM SIGCOMM Conf. Internet Meas. Conf. (IMC)*, 2011, pp. 243–258.
[8] K. Thomas, D. McCoy, C. Grier, A. Kolcz, and V. Paxson, "Trafficking fraudulent accounts: The role of the underground market in Twitter spam and abuse," in *Proc. 22nd USENIX Secur. Symp.*, 2013, pp. 195–210.
[9] M. Coccoli, L. Caviglione, and A. Merlo, "A taxonomy-based model of security and privacy in online social networks," *Int. J. Comput. Sci. Eng.*, vol. 9, no. 4, pp. 325–338, 2014.
[10] L. Shi, S. Yu, W. Lou, and Y. T. Hou, "SybilShield: An agent-aided social network-based Sybil defense among multiple communities," in *Proc. IEEE INFOCOM*, vol. 12, Apr. 2013, pp. 1034–1042.
[11] J. Leskovec and C. Faloutsos, "Sampling from large graphs," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2006, pp. 631–636.
[12] M. Al-Qurishi, M. Alrubaian, S. M. M. Rahman, A. Alamri, and M. M. Hassan, "A prediction system of Sybil attack in social network using deep-regression model," *Future Gener. Comput. Syst.*, vol. 87, pp. 743–753, Oct. 2018.
[13] M. Fire, R. Goldschmidt, and Y. Elovici, "Online social networks: Threats and solutions," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 2019–2036, 4th Quart., 2014.
[14] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou, "A walk in Facebook: Uniform sampling of users in online social networks," *CoRR*, vol. abs/0906.0060, 2009. [Online]. Available: http://arxiv.org/abs/0906.0060
[15] D. Mulamba, I. Ray, and I. Ray, "SybilRadar: A graph-structure based framework for Sybil detection in on-line social networks," in *ICT Systems Security and Privacy Protection*. Cham, Switzerland: Springer, 2016, pp. 179–193.
[16] M. Sirivianos, T. Pregueiro, Q. Cao, and X. Yang, "Aiding the detection of fake accounts in large scale social online services," in *Proc. Usenix Conf. Netw. Syst. Design Implement.*, 2012, pp. 197–210.
[17] J. Franklin, "The elements of statistical learning: Data mining, inference and prediction," *Math. Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005, doi: 10.1007/BF02985802.
[18] T.-K. Huang, M. S. Rahman, H. V. Madhyastha, M. Faloutsos, and B. Ribeiro, "An analysis of socware cascades in online social networks," in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, 2013, pp. 619–630.
[19] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech., Theory Exp.*, vol. 2008, no. 10, Oct. 2008, Art. no. P10008.
[20] W. Wei, F. Xu, C. C. Tan, and Q. Li, "SybilDefender: Defend against Sybil attacks in large social networks," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1951–1959.
[21] J. Wu and Z. Chen, "Human activity optimal cooperation objects selection routing scheme in opportunistic networks communication," *Wireless Pers. Commun.*, vol. 95, no. 3, pp. 3357–3375, Aug. 2017.
[22] J. Wu and Z. Chen, "Data decision and transmission based on mobile data health records on sensor devices in wireless networks," *Wireless Pers. Commun.*, vol. 90, no. 4, pp. 2073–2087, Oct. 2016.
[23] Y. Boshmaf, D. Logothetis, G. Siganos, J. Lería, J. Lorenzo, M. Ripeanu, K. Beznosov, and H. Halawa, "Íntegro: Leveraging victim prediction for robust fake account detection in large scale OSNs," *Comput. Secur.*, vol. 61, pp. 142–168, Aug. 2016.
[24] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "The socialbot network: When bots socialize for fame and money," in *Proc. 27th Annu. Comput. Secur. Appl. Conf. (ACSAC)*, 2011, pp. 93–102.
[25] S. Lv, Y. Zhang, and D. Fan, "Anomaly detection in online social networks," *Chin. J. Comput.*, vol. 38, no. 10, pp. 2011–2027, 2015.

QINGQING ZHOU received the bachelor's degree from the Software College, Jishou University, Zhangjiajie, China, in 2015, and the master's degree from the School of Software, Central South University, Changsha, China, in 2018. She is currently pursuing the Ph.D. degree with the College of Computer Science and Electronic Engineering, Hunan University. Her research interests include machine learning, data mining, social networks, and industrial big data.

GUO CHEN received the bachelor's degree in electronic information school from Wuhan University, Wuhan, China, in 2011, and the Ph.D. degree from the Department of Computer Science from Tsinghua University, Beijing, China, in 2016. He worked with Microsoft Research Asia as an Associate Researcher, from 2016 to 2018. He is currently a Full Associate Professor with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. He research interests include broadly in computer networking and networked system, with a special focus on cloud networking for now. He also does research on AI systems and big data.