Graph Based Local Risk Estimation in Large Scale Online Social Networks

Naeimeh Laleh, Barbara Carminati, and Elena Ferrari
Department of Theoretical and Applied Science
University of Insubria
Varese, Italy
E-mail: {n.laleh, elena.ferrari, barbara.carminati}@uninsubria.it

***Abstract*—Online Social Networks (OSNs) have become extremely popular in recent years, leading to the presence of huge volumes of users' personal information on the Internet. This increases the need for efficient and effective measures helping users to judge their direct contacts so as to avoid friendship with malicious users that could misuse their personal information.**

At this purpose, in this paper we propose a risk measure, called *local risk factor*, having as a key idea the fact the malicious users in OSNs (aka attackers) show some common features on the topology of their social graphs that is different from those of legitimate users. This consideration brought us to design a set of graph based features defined based on attacker activity patterns. To prove the effectiveness of the proposed risk measure, we run several experiments on a real OSN dataset (i.e., Orkut social network) with more than 3 million vertices and 117 million edges, by injecting synthetic fake users according to different settings and showing how the proposed measures can indeed help in their detection.

***Keywords*-Online Social Network (OSN); Risk assessment; Anomaly detection; Graph analysis.**

I. Introduction

It is a matter of fact that Online Social Networks (OSNs) are now part of everyday life for millions of people. These are used to keep in touch with family, friends, share personal information, as well as manage business relationships. As such, OSNs have become a huge collector of personal data, usually shared among network participants according to their connections. As examples, both Facebook and Google+ support sharing rules (i.e., privacy settings) making a user able to state which connections one has to have in order to access his/her personal information. In general, the connection is specified in terms of the relationship type (e.g., groups in Facebook and circles in Google+) and its distance (e.g., direct contacts, friend of friends). These rules greatly simplify information sharing in OSNs, but they bring a serious drawback in that the establishment of a new relationship might imply the exposure of your personal data to a huge amount of unknown person, as an example in case of friend-of-friends sharing rule. This scenario gets even worse if we consider that typical OSN user is now used to establish relationships with people they do not know. Moreover, this is further exacerbated by the fact that the increasing popularity of OSNs has recently encouraged attackers to develop different techniques to exploit OSN infrastructures for malicious purposes. The most notable types of attacks in OSNs are sybils/socialbots, identity cloned attacks, socwares, compromised account attacks, cyberbulling attacks, and creepers [1], [2].

To cope with emerging security and privacy concerns, research community has started to deeply investigate and propose mechanisms for safer and trustworthy OSNs. Orthogonal to all these efforts, we believe there is the need of measures helping user to judge his/her contacts. This brings us to investigate a risk estimation by considering the topology of user's social graph. In doing this, we leverage on relevant results achieved on graph-based outlier detection. In particular, the proposed risk measure comes from the observation done in [3], where it has been highlighted that a user of a given network is anomalous if his/her subgraph significantly differs from those of other users. More precisely, in this paper, we adapt the definitions proposed in [4] so as to have an unsupervised graph-based outlier detection methods tailored over features meaningful for the detection of risk behaviors in OSNs. The overall purpose is to obtain a local risk factor measure that helps users to detect potential attackers among their contacts. The obtained results show that these topological features can indeed be used to define the risk of direct contacts in large scale OSNs.

The remainder of this paper is organized as follows. Section II introduces the overall idea underlying our approach, whereas Section III provides a summary of the considered graph based features. Section IV illustrates our graph based risk measure. Experiments are presented in Section V, whereas related work are discussed in Section VI. Finally, Section VII concludes the paper.

II. Overall Approach

The proposing risk estimation measure comes from the observation done in [3], where it has been highlighted that a user of a given network is anomalous if his/her subgraph significantly differs from those of other users. To define the subgraph, we borrow the terminology from social network analysis (SNA), where "ego" is an individual user in OSNs. In SNA, the direct subgraph of an ego node is known as its "egonet". In our proposal, based on the behavior of malicious users in OSNs, we consider not only the direct subgraph of ego, but also the direct subgraph of his/her direct contacts (two step subgraph). Therefore, the subgraph of an ego is a

collection of all direct contacts of ego, the direct contacts of his/her direct contacts as well, and all the connections among them (see Section III-B for a more detailed discussion).

We have then to define a measure for comparing different subgraphs. We exploit the lesson learnt from an outlier detection technique presented in [5], saying that density can represent an interesting measure to catch anomalous nodes. In particular, by comparing the local density of a node to the local densities of its nearest users, one can identify regions of similar density, and nodes that have a substantially lower density than their nearest users, considered to be outliers. The local density of a user u is computed based on a distance measure. In particular, we first compute the Euclidean distance between u and all the users in the network. Then, we rank the results and we select the distance of the user at the *k-th* position. Thus, local density of u is defined as the inverse of this distance.

Given the purpose of this paper, i.e., risk estimation, we compute the Euclidean distances based on a set of topological features that we believe are meaningful for the detection of risky behaviors. These features have been selected based on a review of current well-known OSN attacks (see discussion in Section III for more details).

According to [4], we can exploit the local density of u, to determine its *Divergency Factor*, that is, how much u is different from the rest of the network. More precisely, how much u's density is different from the ones of users that are topologically similar to u. These users are defined as *k-nearest users*[1].

Literature offers different ways to compute the divergency factor, see for instance [4], [5]. In this paper, we exploit the method proposed in [4], called INFLO. The benefit of this method is that it also considers the symmetric neighborhood relationship in computing the k-nearest users. This means that a user z is into the k-nearest users of a user y if the Euclidean distance between z and y is less than the one at the k-th position in the ranking and there is symmetry in their k-nearest users, that is, y is into the k-nearest users of z as well.

Once the divergency factors for all users in the network have been computed, a target user u will be able to assign a *Local Risk Factor* to each node y in his/her direct contacts list, based on how much y's divergency factor is different from the divergency factor of the other u's direct contacts. Thus, u can understand how much his/her contacts are risky by ranking their local risk factors (see Section IV for more details).

III. Topological-based Features for Risk Estimation

In this section, we introduce the features we use for computing our measures. We have driven the selection of them by what have been so far recognized as risky users in OSNs, that is, attackers. The topological patterns of attackers are, in general, different from those of normal users. The discrepancy is defined in terms of the structure of their social graphs which are hard to be changed by attackers. In the following, we first summarize the most notable attacks and related topological information, we then introduce the considered features.

A. Risky behaviors in OSNs

Sybil attacks are one of the most prevalent and practical attack in OSNs [6]. To launch a Sybil attack, a malicious user has to create multiple fake identities, known as Sybils, with the purpose to legitimate his/her identity [6], so as to unfairly increase his/her power and influence within a target community. After that, attackers start sending friendship requests to other users in the community. Once the requests have been accepted, the socialbot can gather users' private data. Sybil attacks can be classified into three main types.

The first is Sybils with a *tight-knit community* (dense friendship graph), where adversaries create huge number of Sybils by also establishing connections among them [7], [8]. Due to this high number of connections, Sybils tend to form tight knit clusters in their direct subgraph.

In contrast, authors in [9] have analyzed the distribution of Sybil accounts in the Renren OSN[2]. This shows that the vast majority of Sybil accounts do not form social links among them. Moreover, even in case they form, the resulting clusters are loose, rather than tightly connected. They found that attackers use snowball sampling techniques to identify and send friend requests to popular users, since these are more likely to accept requests from strangers. Therefore, Sybils have friendship links with a lot of strangers and their friendship graph become sparse. As such, the analyzed Sybils shows to belong to another type of Sybils attackers, that is, Sybils with a *sparse community* (sparse friendship graph). Moreover, authors in [3] show that the majority of anomalous users in an OSN have neighbors that are either very well connected (forming a near-cliques), similar to Sybils with tight-knit community, or not connected (stars), similar to Sybils with sparse community.

In Sybils with sparse friendship graph, attackers after creating a huge number of Sybil accounts establish few connections among themselves, and then they try to send friendship requests to popular users.

In addition, some works show that Sybils first establish few connections among themselves, and then they try to send friendship requests to other users. Researchers prove that in most cases Sybils fail to create friendship links with legitimate users [7], [8]. In this way we introduce the third category of Sybils with a friendship graph that is not sparse or dense in the direct subgraph, since they have a small number of mutual friends with their direct contacts. Although Sybils in this category have a normal friendship graph in their direct subgraph, but they fail to create friendship links with legitimate users and majority of their friends are popular users

[1]Given a k, the *k-nearest users* are determined by computing the Euclidean distance of u with all other users in the network. Once all the Euclidean distances of u with other users in the network have been computed, we rank the results and we select the value $dist_k(u)$ of the Euclidean distance at the k-th position in the ranking. Based on this value, we can define the *k-nearest users of a target user* u as the set of users whose Euclidean distances with u is less than $dist_k(u)$.

[2]http://www.renren-inc.com/en/

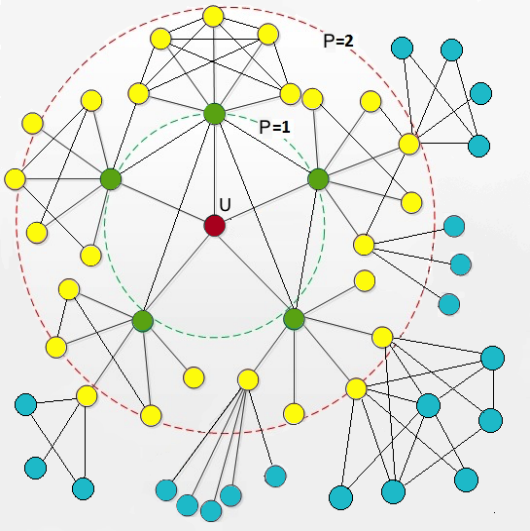


Fig. 1: Subgraph for each target user

or other malicious users. Therefore, the structure of the direct subgraph of their friends is different from those of legitimate users.

A final note is about the generation of fake profiles, which are created by non-malicious users, called creepers [1] wishing some extra accounts, for several purposes, like social reasons such as friendly pranks, stalking, cyberbullying, etc. [1], [8]. A recent paper stated that the market of buying fake followers and fake retweets is already a multimillion-dollar business [10].

B. Features

Given a user u, we model the graph from which the topological features of u are extracted as the subgraph formed by the set of users, and related relationships, that can be reached by at maximum P-steps from u as exemplified by Figure 1. In computing the features, we consider P=2, since, as discussed in [11], real social networks show a small diameter. Based on the discussion in Section III-A, we consider the following topological features of u, extracted from its 2-step subgraph:

- Degree of u, ($Degree$), that is, the number of direct contacts of u;
- Triangles count of u, ($TriangleCount$), where a triangle exists when a node has two adjacent nodes that are also adjacent to each other;
- The ratio between degree and triangle count of u, that is, $RateDT = Degree/TriangleCount$.
- The average degree of all direct contacts of u, ($AvgDegree$);
- The average triangle counts of all direct contacts of u, ($AvgTriangleCount$);
- The average ratio between degree and triangle count of all direct contacts of u, $AvgRateDT = Avg(Degree/TriangleCount)$.

IV. Local Risk Score

Our goal is to assign a risk score to the direct contacts of a target OSN user u, based on the deviation of their divergency factors. As introduced in Section II, we exploit the Influence Outlierness (INFLO) [4] for the computation of the divergency factor. INFLO exploits not only the k-nearest users, but also, the reverse k-nearest users (RNU) [12]. Members of RNU of a user u are users that have u as one of their k-nearest users. More formally, we introduce the definition of k-nearest users and reverse k-nearest users.

***Definition** 1 (*k-nearest users of u): Let G be the graph modeling the OSN, and u be a node in G. Given a value k, the k-nearest-users of u are defined as:

$$NU_k(u) = \{u' \mid u' \in G, dist(u,u') \leq dist_k(u)\} \quad (1)$$

where $dist(u,u')$ denotes the Euclidean distance between u and u' computed on a selection of features among $\{Degree, TriangleCount, RateDT, AvgDegree, AvgTriangleCount, AvgRateDT\}$;[3] $dist_k(u)$ is the Euclidean distance value between u and the user in G placed in the k-th position w.r.t. the Euclidean distance ranking.

***Definition** 2 (*Reverse k-nearest users of u): Let G be the graph modeling the OSN, and u be a node in G. Given a value k, the reverse k-nearest-users of u is defined as:

$$RNU_k(u) = \{u' \mid u' \in G, u \in NU_k(u')\} \quad (2)$$

Given a user u the union of $NU_k(u)$ and $RNU_k(u)$ forms its local neighborhood space denoted as $SN_k(u)$ to estimate the density distribution around u. According to [4], INFLO is defined as the ratio of the average density of users in $SN_k(u)$ to the u's local density:

$$INFLO_k(u) = \frac{den_{avg}(SN_k(u))}{den(u)} \quad (3)$$

where den_{avg} is the average density of users in $SN_k(u)$ and $den(u)$ is the local density of user u. Based on INFLO, we can now provide the definition of divergency factor.

***Definition** 3 (*Divergency Factor): Let G be the graph modeling the OSN, and u be a node in G. Given a value k, the Divergency Factor of u, $DF_k(u)$, is given by $INFLO_k(u)$, where the Euclidean distances are computed on a selection of features among $\{Degree, TriangleCount, RateDT, AvgDegree, AvgTriangleCount, AvgRateDT\}$.

The divergency factor is a measure to define how much the neighborhood of a given user u is different from his/her nearest users. We can then use the divergency factor to assign a local risk factor to each direct contact of a target user u. This risk measure is defined on the basis of how much a direct contact of u is different from other u's contacts. We refer to this as the divergency factor deviation (DFD), formally defined as follows.

[3]As described in Section V, in order to validate the effectiveness of the proposed features, we carried out experiments by considering not only all the six features, but also a selection of them.

Definition 4 *(*Divergency factor deviation): Let G be the graph modeling the OSN, let u be a node in G, and let y be one of its direct contacts. Given a value k, the divergency factor deviation of y for u is the defined as:

$$DFD_k(u, y) = DF_k(y) - (STDDF(u) + ADF(u)) \quad (4)$$

where $DF_k(y)$ is y's divergency factor; $STDDF(u)$ and $ADF(u)$ are the standard deviation and the mean of the divergency factor values of all direct contacts of u, respectively.

Based on the above definitions, given a target node u and one of its direct contact, say y, we have two divergency measures for y. The first one is $DF_k(y)$, that shows how much the density of the neighborhood of y is different from its density. The second one is $DFD_k(u, y)$, that shows how much the divergency factor of y is different from the divergency factor of other contacts of u. We combine these two measures to obtain the Local Risk Factor, formally defined as follows.

Definition 5 *(*Local Risk Factor): Let G be the graph modeling the OSN, a let u be a node in G, and let y be one of its direct contacts. Given a value k, the Local Risk Factor of y for u is the defined as:

$$LRF_k(u, y) = DF_k(y) + DFD_k(u, y) \quad (5)$$

Given a target node u, we first compute the LRF for each of its direct contacts, then we rank them based on their LRFs and we flag as risky those contacts whose LRF is higher than a threshold, denoted as $LRFT(u)$ and defined based on the distribution of LRF values of u's contacts. In particular, the threshold for target user u is computed as: $LRFT(u) = STDLRF(u) + MeanLRF(u)$, where $STDLRF(u)$ and $MeanLRF(u)$ are the standard deviation and the mean of the LRFs of all direct contacts of u, respectively.

V. Experiments

Experiments aim at showing how the proposed local risk factor measure can be used to detect risky users in the contact list of a target user u. At this purpose, we have used a real social graph, that is, the Orkut Online Social Network (OSN) dataset taken from SNAP [4]. In this dataset, there are 3,072,441 nodes and 117,185,083 edges. Unfortunately, the dataset is not provided with a ground truth, in that we do not have any information about which nodes in the Orkut dataset are risky. This is a common problem in validating anomaly detection techniques, where, as discussed in [13], several different validation approaches have been used in the literature, such as anomaly injections or qualitative analysis. There are several works based on anomaly injection to evaluate the result of anomaly detection models [14]–[16]. In this paper, we follow the idea of injecting fake users into the real graph, that is, nodes and random connections, created such as to simulate some kind of attacks. In particular, we simulate four different categories of risky users in OSNs (see discussion in Section III).

[4]https://snap.stanford.edu/data/

For each category, we inject the nodes into the Orkut dataset. Then, we compute a divergency factor for all the users in the obtained new dataset and the local risk factor for all the injected fake users. The LRF of a fake user is computed w.r.t. a target user u. In particular, the target user is selected among the real nodes in the Orkut social graph that have at least a connection with the injected fake node. Finally, we flag as risky a fake user, if its LRF deviates from those of the other direct contacts of u, based on the computed threshold value. In other words, if the LRF of fake node is higher than a local threshold of target user, we flag him/her as risky.

In all the experiments we set k from 5 to 10, since considering a high value for k in large social graph has a high computational cost and, according to [4], does not have a big effect on the result.

Moreover, in order to test the effectiveness of the features described in Section III-B, we carried out experiments by considering different combinations of the features. In the following, we first describe the feature combinations, we then introduce the three categories of injected fake users, finally we discuss the experimental results.

A. Features settings

Once the six features described in Section III-B have been computed, we consider the following different combinations for computing the Euclidean distances.

All the six features. According to this setting, we consider all the six features described in Section III-B. Figure 2 shows the distribution of computed divergency factors for all users in the considered dataset. In this feature setting, the DF is in the range between [0, 250] and most of the users have a DF in the range [0, 2.7], whereas few of them are with DF higher than 2.7.

Ratio of degree to triangles count. In this setting, we consider only $RateDT$ and $AvgRateDT$ in the divergency factor computation. In the $RateDT$, we are interested to catch those users for witch there is no balance between their degree and triangle count as this could indicate a risky conduct. In addition, by bringing $AvgRateDT$ into account, we consider also for all his/her direct contacts as well. Therefore, we consider both the subgraph around each user and all his/her direct contacts for our risk estimation.

In this way, we are able to catch users whose two steps subgraph are very well connected (near-cliques) or not connected (stars). As we can see in Figure 3, most of the users have a DF in the range [0, 2.7], whereas just few of them diverge from their nearest users with DF higher than 2.7. The range of DF is [0, 84].

B. Injected risky users

In this section, we introduce the three categories of fake users we inject in the Orkut dataset. **Sybils with sparse friendship graph.** The first kind of attack that we try to simulate is Sybils with sparse friendship graph in their direct subgraph. As we discuss in Section III-A, these kind of attackers have friendship links with a lot of strangers by

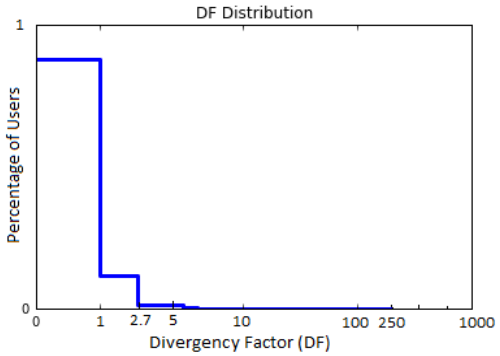


Fig. 2: DF distribution by considering all the six features

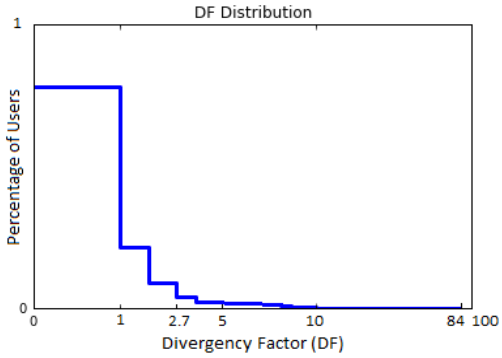


Fig. 3: DF distribution by considering Ratio and AvgRateDT

using random sampling techniques to send friend requests to strangers. Therefore, we inject 100 users, each one having a number of edges selected randomly in the range of the mean and the sum of the mean and the standard deviation of degree of all users in the real graph (i.e., values in [100, 250]) to be similar to regular users. Then, we totally create around 10000 to 25000 friendship links among these 100 Sybils with randomly selected users from the whole graph by considering random sampling.

Sybils with dense friendship graph. The second type of attackers are Sybils with dense friendship graph or tight-nit communities. To model these attackers, after creating 100 fake nodes and inject them into the graph, we generate the edges among themselves and then, with a set of randomly selected users and also the 80% of their direct contacts to have more mutual friends with each friend. Moreover, we generate these edges so that each fake node has a degree in the average range of all other legitimate users, to be more similar to other regular users.

Sybils with normal friendship graph. The third type of attackers are Sybils with normal friendship graph. In this kind of attacks, attackers after creating a huge number of Sybil accounts establish few connections among themselves, and then they try to send friendship requests to popular users. To model these attackers, after creating 100 fake nodes and inject them into the graph, we generate the edges with a set of randomly selected users with high degree in the range of [1000, 33000]. Then, totally we create around 10000 to 25000 friendship links among these 100 Sybils with popular users.

Real users with additional fake accounts (creepers). These risky users are real users wishing some extra accounts. Usually these users have not a high degree in the graph, but they just create a fake account and then, randomly pick us some strangers and make friendship links with these random users. The difference of these creepers with Sybils with sparse friendship graph is that they have few friendship links since their goal is not as attackers to influence the graph by having more links. To model this type of risky users, we inject 100 users each having a number of edges in the range of the mean minus standard deviation and mean of degree of all users in real graph (i.e. values in [50, 150]). We create these friendship link with randomly selected users from whole the graph. Then, totally we create around 5000 to 150000 friendship links among all these 100 fake users and other users in the network.

C. Experimental results

We run our experiment with two different feature settings previously discussed on the four different graphs with injected risky users. After calculating the LRF for each user, we compute the difference of the LRF value with the local LRF threshold of target user $LRFT(u)$ that is in contact with the risky user. In this way, if the LRF of each user is higher than $LRFT(u)$ among the other contacts of target user u, the user is detected as risky. We consider the value zero for those risky users such that their LRF is lower than the $LRFT(u)$, since they are not deviate from other contacts of target user u, that is:

$$\begin{cases} LRF_k(u,y) - LRFT(u) & \text{if } LRF_k(u,y) > LRFT(u) \\ 0 & \text{if } LRF_k(u,y) \leq LRFT(u) \end{cases}$$

We flag as risky those users with the difference of their LRF and $LRFT(u)$ higher than zero. The result of the first and second feature settings for the four categories of risky users is shown in Table I. Here, we can see the percentage of risky users that are detected by the majority (more than 50%) of target users that are in contact with them. Furthermore, Table II represents the percentage of fake users that are detected as risky by at least one of the target users that are in contact with them. In more detail, Figure 4 shows 100 different categories of fake users that are detected as risky with the percentage of target users that are in contact with each one. In particular, the x-axis shows the percentage of target users that are in contact with each fake user and able to detect him/her as risky and the y-axis show the percentage of fake users that are detected as risky. As we can see in the Figure 4, most of the fake users are detected with more than 50 % ($>=$ 0.5 in x-axis) of target users that are in contact with them. Figure 5 shows all target users that are in contact with 100 Sybils with sparse friendship graph, to see how many of the target users are able to detect these 100 Sybils as risky. As we can see in Figure 5, among 15490 target users that are in contact with these Sybils, around 13470 are able to detect these Sybils in their contact

Feature Setting	Sparse Sybils	Dense Sybils	Sybils with normal direct subgraph	Risky Fake Accounts (Creepers)
All the Six Features	90%	41%	79%	77%
RateDT and AvgRateDT	95%	76%	90%	95%

TABLE I: Detection rate of risky users detected by majority of the target users

Feature Setting	Sparse Sybils	Dense Sybils	Sybils with normal direct subgraph	Risky Fake Accounts (Creepers)
All Six Features	100%	52%	95%	89%
RateDT and AvgRateDT	100%	99%	99%	98%

TABLE II: Detection rate of risky users detected by of at least one of the target users

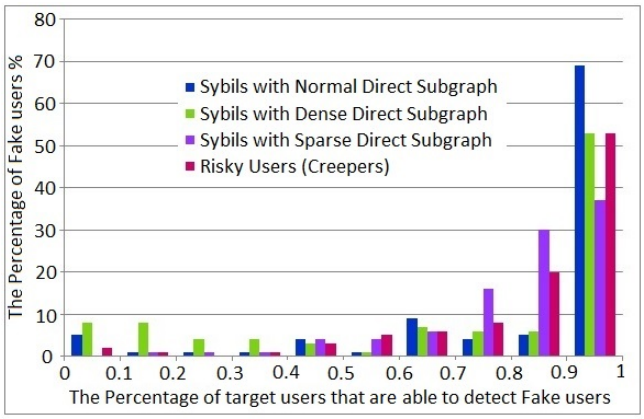


Fig. 4: Risky users that are detected with target users in feature setting (RateDT and AvgRateDT)

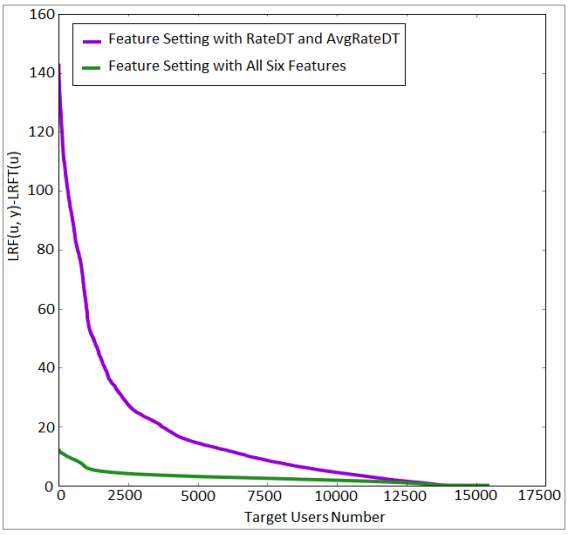


Fig. 5: Target users that are able to detect Sybils with sparse direct subgraph

list as risky that is around 86.95 percent of all target users that are in contact with them. To calculate the performance of our risk models with different feature settings, we compute the F-measure. We need to mention that for calculating F-measure, we need to compute precision and recall that is based on: false positive (FP), false negative (FN), true positive(TP) and true negative (TN). In order to calculate precision and recall, we need to calculate also the false positive that is the number of legitimate users that are detected as risky in our risk models. Therefore, the evaluation based on precision and recall is challenging since it would be severe to call the risky users detected other than the injected ones as false positives, given that the original real graph may also contain same type of anomalies and risky users [13].

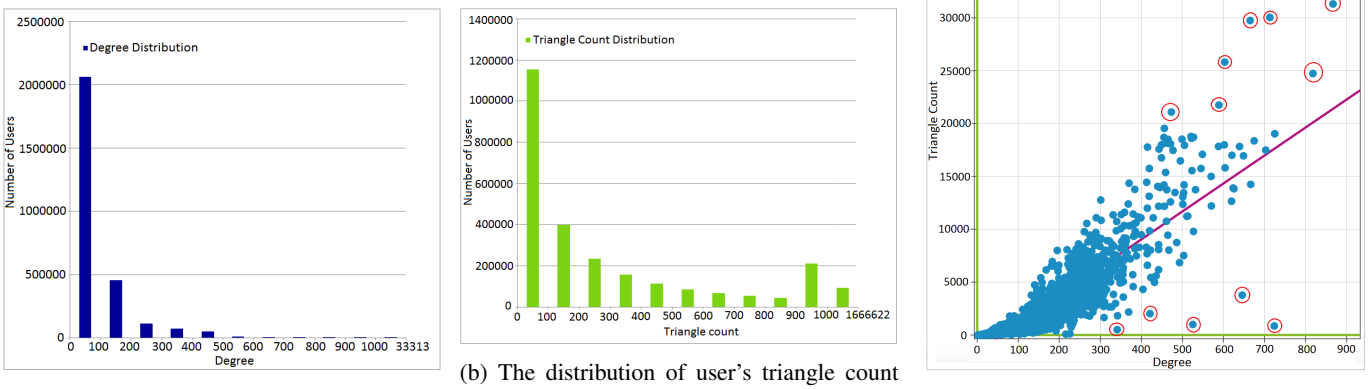
Based on anomaly detection concept, the majority of users obey a pattern and only few users that deviate, considered as outliers [3]. Therefore, to consider a set of legitimate users, we selected 1000 legitimate users randomly not from the whole graph but among those users that their graph structure is similar to majority of users inside the real graph. In other words, we didn't consider outliers for this selection.

Then, we find a measure to find legitimate users since considering all users with high degree or low degree and also users with high triangle count or low triangle count as legitimate or anomalous is not reasonable. This is motivated because, there is a large number of popular users with high degree as shown in Figure 6a that is around 30,105 users with number of degree higher than 700 in the range of [700, 33313]. Figure 6a shows the number of users in the x-axis and the number of degree in y-axis. The maximum degree of users in the graph is 33313. Furthermore, there is a hight number of users with high triangle count or isolated users with low triangle count as shown in Figure 6b. The maximum value of triangle count is 1,666,622 that we can see there are around 100,000 users with triangle count in the range of [1000, 1666622].

But, Figure 6c shows the relation of increasing the degree with triangle count that is RateDT for all users inside the real social graph. As we can see the majority of users have a RateDT near the red line and just few users surrounded with red circles have these values outside the line that is considered as outlier in [3]. Therefore, we select our 1000 normal users randomly among those users with RateDT between 0.1 and 10, since 98% of users have this range of values.

Then, after computing their LRF, we consider all target users that are in contact with normal users to see the percentage of these normal users that by mistake are detected as risky. For example, between all target users (33156) that are in contact with the 1000 normal users when we consider (RateDT and AvgRateDT) as features, around 1196 of them detect these normal users as risky that is 3.60% percent of all target users.

Table III represents the F-measure for the two feature settings with all four categories of fake users when the majority



(a) The distribution of user's degree in OSN

(b) The distribution of user's triangle count in OSN

(c) The plot of user' RateDT in OSN

Fig. 6: The distribution of user's degree, triangle count and RateDT in Orkut OSN

Feature Setting	Sparse Sybils	Dense Sybils	Sybils with normal direct subgraph	Risky Fake Accounts (Creepers)
All Six Features	0.90	0.54	0.839	0.826
RateDT and AvgRateDT	0.939	0.821	0.925	0.936

TABLE III: F-measure in the two feature settings (majority of target users detect risky users)

Feature Setting	Sparse Sybils	Dense Sybils	Sybils with normal direct subgraph	Risky Fake Accounts (Creepers)
All Six Features	0.955	0.649	0.93	0.897
RateDT and AvgRateDT	0.961	0.956	0.95	0.951

TABLE IV: F-measure in the two feature settings (At least one target user detect risky users)

of the target users, in contact with them, have detected them. Based on the result, the performance of risk model with two feature settings (RateDT and AvgRateDT) is the best, since these are the most influential features that reveal these kind of risky structural patterns in the graph. As we can see, the performance of detecting sybils with sparse direct subgraph is the best around 90% and sybils with normal direct subgraph and creepers are in the second rank still more than 90%. The performance of detecting sybils with dense direct subgraph is lower around 82% since there are some other users inside the real graph with severe case than this category of sybils with a very high degree more than 1000 and very high triangle count more than 900,000 that make their direct subgraph denser than Sybils. Also, Table III shows the value of F-measure when at least one target user detects fake users as risky. Based on the result, again the performance of risk model with two feature settings (RateDT and AvgRateDT) is the best. Also, the performance of all four categories of fake users are more than 95 % that is a very good result. We can see that our risk model is able to help target users to detect risky users with a high accuracy and low false alarm (FP) rate.

VI. Related Work

Relevant for our proposal are the works targeting graph-based outlier detection (see [13] for a survey). Among them, structure-based approaches make use of graph-centric features, such as node degree and subgraph centrality [17], that are sometimes used together with other features extracted from additional information sources to identify outliers. The feature-based approaches have been used in several anomaly detection application domains, including network intrusion detection [18], web spam detection [19] and, fraud detection [20].

On of the research works, ODDBALL [3] extracts ego network features by considering one step subgraph, such as the degree, total weight, principal eigenvalue, etc. to find patterns that most of the nodes of the graph follow with respect to those features and spot anomalous nodes as those that do not follow the observed patterns. In our approach we consider different features driven by the structural behavior of attackers in real OSNs. In addition, we considered the 2 step subgraph (the network of all direct contacts of ego). Because based on the behavior of attackers in OSNs, considering only ego network features is not enough. More precisely, researchers stated that most sybils and fake accounts can not create link with normal users and most of their friends are either sybils or popular users [21]. Therefore, considering the network of direct contacts of attackers is important to reveal these kinds of structural behavior. Another research work use recursive graph based features to capture behavioral information for classification and de-anonymization tasks without the availability of class labels [22], although their goal is not anomaly detection.

One of the application of anomaly detection in OSN is spam filtering. [23] performs online spam filtering on social networks using incremental clustering, based on network-level features such as sender's degree and the interaction history between users.

In addition of anomaly detection area, there are some approaches for Sybil detection [7], [8]. These approaches uses different graph analysis algorithms to search for legitimate and Sybil users. Although, these schemes work by analyzing the structure of the social network, all of them make three common assumptions. First, the legitimate region of the graph is densely connected. Second, attackers cannot establish a high number of social connections to legitimate users. Third, the system is given the identity of at least one legitimate user. Thus, the performance of these schemes is heavily dependent on the size and characteristics of the community surrounding the legitimate users. Furthermore, another approach is [24] that is a combination of graph and content based to detect Sybils. All of the above mentioned approaches are supervised.

As the risk assessment in OSN is concerned, [25] propose a measure for risk estimation by considering the profile similarity and number of mutual friends that a target user has with other strangers as a measure that how much is risky to become friend with a stranger. They used supervised classification to assign a risk score. However, due to the challenges in obtaining labels, supervised learning algorithms are less attractive for the task of risk assessment. In our proposal, we focus on risk assessment in online social networks based on unsupervised graph based anomaly detection.

VII. Conclusion

In this paper, we propose a local risk estimation measure (Local Risk Factor) for direct contacts of a target user. Our risk estimation is based on anomaly detection algorithm having as key idea the fact the malicious users in OSNs show some common features on the structure of their social graphs that is different from those of legitimate users. We demonstrate that LRF can be used to define the risk of direct contacts efficiently in large scale OSNs. We also show that some of the features are more influential in risk assessment in OSN than others. For future directions we will use other user features to define risk score and develop more accurate models for risk assessment in OSNs. In addition, graph based approaches for risk assessment in dynamic graph is challenging task and we will apply them to have a high performance risk models that is more robust in online social network by changing the behaviour of attackers.

Acknowledgment

This work is supported by the iSocial EU Marie Curie ITN project (FP7-PEOPLE-2012-ITN).

References

[1] T. Stein, E. Chen, and K. Mangla, "Facebook immune system," in *Proceedings of the 4th Workshop on Social Network Systems*. ACM, 2011, p. 8.

[2] M. Fire, R. Goldschmidt, and Y. Elovici, "Online social networks- threats and solutions," 2013.

[3] L. Akoglu, M. McGlohon, and C. Faloutsos, "Oddball: Spotting anomalies in weighted graphs," in *Advances in Knowledge Discovery and Data Mining*. Springer, 2010, pp. 410–421.

[4] W. Jin, A. K. Tung, J. Han, and W. Wang, "Ranking outliers using symmetric neighborhood relationship," in *Advances in Knowledge Discovery and Data Mining*. Springer, 2006, pp. 577–593.

[5] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *ACM sigmod record*, vol. 29, no. 2. ACM, 2000, pp. 93–104.

[6] C. A. Freitas, F. Benevenuto, S. Ghosh, and A. Veloso, "Reverse engineering socialbot infiltration strategies in twitter," *arXiv preprint arXiv:1405.4927*, 2014.

[7] Y. Boshmaf, K. Beznosov, and M. Ripeanu, "Graph-based sybil detection in social and information systems," in *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on*. IEEE, 2013, pp. 466–473.

[8] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services." in *NSDI*, 2012, pp. 197–210.

[9] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai, "Uncovering social network sybils in the wild," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 8, no. 1, p. 2, 2014.

[10] G. Stringhini, G. Wang, M. Egele, C. Kruegel, G. Vigna, H. Zheng, and B. Y. Zhao, "Follow the green: growth and dynamics in twitter follower markets," in *Proceedings of the 2013 conference on Internet measurement conference*. ACM, 2013, pp. 163–176.

[11] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, and S. Vigna, "Four degrees of separation," in *Proceedings of the 4th Annual ACM Web Science Conference*. ACM, 2012, pp. 33–42.

[12] Z. Chen, A. W.-C. Fu, and J. Tang, "On complementarity of cluster and outlier detection schemes," in *Data Warehousing and Knowledge Discovery*. Springer, 2003, pp. 234–243.

[13] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: a survey," *Data Mining and Knowledge Discovery*, vol. 29, no. 3, pp. 626–688, 2014.

[14] H. Dai, F. Zhu, E.-P. Lim, and H. Pang, "Detecting anomalies in bipartite graphs with mutual dependency principles," in *Data Mining (ICDM), 2012 IEEE 12th International Conference on*. IEEE, 2012, pp. 171–180.

[15] A. B. Sharma, H. Chen, M. Ding, K. Yoshihira, and G. Jiang, "Fault detection and localization in distributed systems using invariant relationships," in *Dependable Systems and Networks (DSN), 2013 43rd Annual IEEE/IFIP International Conference on*. IEEE, 2013, pp. 1–8.

[16] W. Eberle and L. Holder, "Compression versus frequency for mining patterns and anomalies in graphs," in *Ninth workshop on mining and learning with graphs (MLG 2011), SIGKDD, at the 17th ACM SIGKDD conference on knowledge discovery and data mining (KDD 2011)*, 2011.

[17] K. Henderson, T. Eliassi-Rad, C. Faloutsos, L. Akoglu, L. Li, K. Maruhashi, B. A. Prakash, and H. Tong, "Metric forensics: a multi-level approach for mining volatile graphs," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 163–172.

[18] Q. Ding, N. Katenka, P. Barford, E. Kolaczyk, and M. Crovella, "Intrusion as (anti) social communication: characterization and detection," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 886–894.

[19] L. Becchetti, C. Castillo, D. Donato, S. Leonardi, and R. A. Baeza-Yates, "Link-based characterization and detection of web spam." in *AIRWeb*, 2006, pp. 1–8.

[20] N. Laleh and M. Abdollahi Azgomi, "A hybrid fraud scoring and spike detection technique in streaming data," *Intelligent Data Analysis*, vol. 14, no. 6, pp. 773–800, 2010.

[21] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "Design and analysis of a social botnet," *Computer Networks*, vol. 57, no. 2, pp. 556–578, 2013.

[22] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos, "It's who you know: graph mining using recursive structural features," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 663–671.

[23] H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. N. Choudhary, "Towards online spam filtering in social networks." in *NDSS*, 2012.

[24] Y. Boshmaf, D. Logothetis, G. Siganos, J. Lería, J. Lorenzo, M. Ripeanu, and K. Beznosov, "Íntegro: Leveraging victim prediction for robust fake account detection in osns," in *Proc. of NDSS*, 2015.

[25] C. G. Akcora, B. Carminati, and E. Ferrari, "Privacy in social networks: How risky is your social graph?" in *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*. IEEE, 2012, pp. 9–19.