TrustGCN: Enabling Graph Convolutional Network for Robust Sybil Detection in OSNs

Yue Sun
Computer Science Department
Peking University
Beijing, China
sy_sunyue@pku.edu.cn

Zhi Yang*
Computer Science Department
Peking University
Beijing, China
yangzhi@pku.edu.cn

Yafei Dai
Computer Science Department
Peking University
Beijing, China
dyf@pku.edu.cn

***Abstract*—Detecting fake accounts (also called Sybils) is a fundamental security problem in online social networks (OSNs). Existing feature-based or social-graph-based approaches suffer from the key limitations: they can only leverage either node feature or graph structure properties such as fast-mixing and conductance, but not both. To overcome this shortcoming, we explore the introduction of recent advancements in deep neural networks for graph-structured data into Sybil detection field. These types of models enable integrating both user-level activities and graph-level structures for a new generation of feature-and-graph-based detection mechanisms. However, we find that although applying Graph Convolutional Networks (GCNs) are effective against naïve attacks, they are vulnerable to adversarial attacks in which fake accounts alter local edges and features with patterns to resemble real users.**

In this paper, we present TrustGCN, a Sybil-resilient defense algorithm that combines the idea of social-graph-based defense with GCN. TrustGCN first assigns trust scores to nodes based on the landing probability of short random walks that starts from known real accounts. As this short, supervised random walk is likely to stay within the subgraph consisting of real accounts, most real accounts receive higher trust scores than fakes. Then it introduces these trust scores as edge weights and adopts graph convolution operations to aggregate features of local graph neighborhoods over this weighted graph for classification. In this way, we prevent Sybil partners with low trust scores from contributing to the feature aggregation for a target node, thus is more robust against adverse manipulations of the attackers. Our experiment on real data demonstrates that TrustGCN significantly outperforms GCN in the robustness. To the best of our knowledge, this is the first attempt to combine social-graph-based defenses with graph neural networks into a unified model, paving the way for the robust feature-and-graph-based detection mechanisms.

***Index Terms*—Sybil detection, graph convolutional network, adversarial attacks**

I. Introduction

Sybil defense is one of the primary concerns of distributed systems [1]. Sybils, created as fake identity accounts, aim to exert or spread malicious influence on the systems. And they are regarded as big threats to the security of P2P systems [2], recommendation systems [3], [4], and anonymous communication platforms [5]. Nowadays, online social networks (OSNs) become hot targets of Sybil attacks [6]. The popular coverage and rapid propagation of information on OSNs offer great opportunities and profits for Sybils. The attackers can spread out spams, scams, or malware in a quick and wide-effecting way with Sybils in OSNs. Detecting Sybils is one of the most important tasks in social network, which can improve the experience of its users and their perception of the service by stemming annoying spam messages and invitations.

Most of today's fake account detection mechanisms are either feature-based or graph-based, depending on whether they utilize machine learning or graph analysis techniques to identify fakes. In the feature-based approach, unique features are extracted from recent user activities (e.g., frequency of friend requests, fraction of accepted requests) and local graph structure, after which they are applied to a classifier that has been trained offline using machine learning techniques. In the social-graph-based approach [7]–[10], also called random walk-based solutions, an OSN is formally modeled as a graph, and the approach relies on social graph properties to uncover fake users: Given assumption that the number of ties that the adversary can forge between Sybils and honest nodes is restricted, the trust propagation (via short random walks) from the normal users would hardly reach Syibls.

These mechanisms, however, suffer from the key limitation: they only leverage either node features or graph structure, but not both, often leading to unsatisfied detection accuracy. For example, feature-based detection is still relatively easy to circumvent, whereas most social-graph-based approaches are used to rank users instead of classification due to relatively high false positive rate.

Motivation. Recent years have seen significant developments of new deep learning methods that are capable of learning on graph-structured data. Most prominent among these recent advancements is the success of Graph Convolutional Networks (GCNs) [11], [12] and their variants, which have become the de facto methods in many supervised and semi-supervised graph representation learning scenarios such as node classifications. Through iteratively aggregating feature information from local graph neighborhoods using deep neural networks, GCNs allow to incorporate both graph structure as well as node feature for the classification. Therefore, different from traditional machine learning or graph analysis techniques,

*Corresponding author.

IEEE/ACM ASONAM 2020, December 7-10, 2020
978-1-7281-1056-1/20/$31.00 © 2020 IEEE

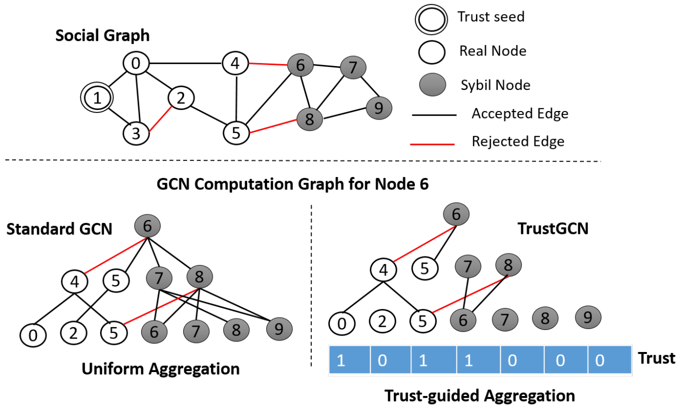
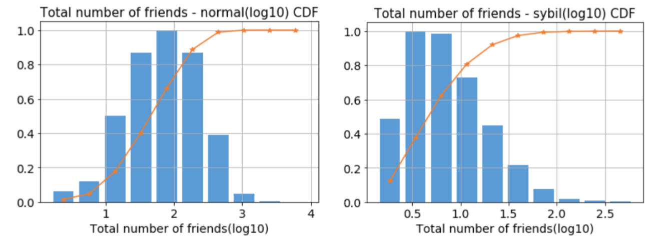
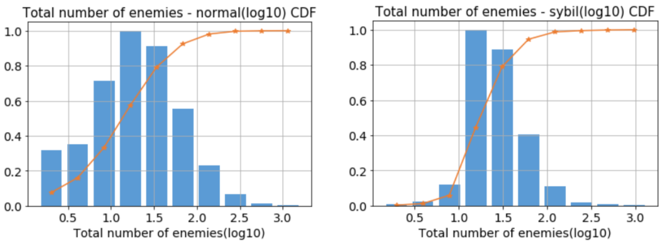


Fig. 1. The key difference between standard GCN and TrustGCN.



(a) Normal (b) Sybil

Fig. 2. The number of accepted friend requests of real nodes and Sybils.



(a) Normal (b) Sybil

Fig. 3. The number of rejected friend requests of real nodes and Sybils.

GCNs would open a promising direction for designing new feature-and-graph-based Sybil detection mechanism.

Challenge. To exploit this direction, we study the performance of GCN in a real dataset where fake accounts perform malicious befriending activities. We model the friend request interactions among users as a signed graph, with the sign indicating the acceptance/rejection. We find that although GCN could achieve high detection accuracy against naïve attacks, it is vulnerable to adversarial attacks and can be easily evaded if fake accounts manipulate their local graph structure.

The key reason is that GCN mainly focuses on the local graph neighborhood of a target node. Thus, it is possible for the attacker to manipulate the links of its local neighbors to affect the GCN's performance. As illustrated in Figure 1, Sybils can send/accept requests among themselves. As standard GCN allows each node in the neighborhood contributing to the feature aggregation of the target node, the manipulation from colluding fake neighbors could lead to significant accuracy drop for GCNs. In this paper, we aim to tackle the question: "How can we design a Sybil-resilient defense that allows to leverage GCNs for the new generation of feature-and-graph-based Sybil detection mechanism?".

Our Solution. We introduce *trust* as a way to improve the robustness of GCN. The key intuition is that, due to the difficulty of soliciting and maintaining reciprocal social relationships, Sybils have more rejected friend requests than accepted ones for signed friend request graph, or establish a limited number of attack edges to real users for unsigned social graph. The low fraction of accepted links (or limited attack edges) result in a negative cut (or a sparse cut) between the non-Sybil and the Sybil region.

To exploit such structural gap for robust detection, we present TrustGCN, which combines GCN with the idea of social-graph-based defenses. Specifically, TrustGCN decouples the feature aggregation into two stages: At the first stage, TrustGCN computes trust scores of nodes in the signed friend activity graph (or unsigned social graph) based on the land probabilities of short random walks starting from known benign nodes. Given the structure gap between Sybils and real users, most real accounts would have higher trust than Sybils. At the second stage, TrustGCN integrates these trust scores of nodes into the graph convolution operation as weights (or the probability for selection), in order to guarantee that most neighbors for feature aggregation are selected from those real users. As illustrated in Figure 4, TrustGCN could effectively prevent most distrust Sybil partners from contributing to the neighborhood feature aggregation of a specific fake node, irrespective of manipulated local neighborhood.

Notice that even the attacker could manipulate the node-level features of individual Sybils (e.g., the fraction of accepted requests), the local neighborhood feature is hard to manipulate as most Sybils cannot effectively contribute to the neighborhood feature aggregation. This enables TrustGCN to be attack-resilient, overcoming the limitations of both existing feature-based and social-graph-based approaches.

Contributions. The key contribution of this paper is that, for Sybil detection in OSN, we show that GCNs are vulnerable to adversarial local structure manipulation, and advocate combining social-graph-based defenses and graph neural networks for robust detection. Given various types of social-graph-based methods and graph neural networks, TrustGCN represents a critical step in this direction by showing not only the feasibility, but also the potential of such combination.

II. Resilience of GCNs

In this section, we examine the resilience of GCN under different attacking strategies.

A. Graph Construction

Sybil Dataset. We use a regional Peking university (PKU) network of complete friend request records from Renren [13].

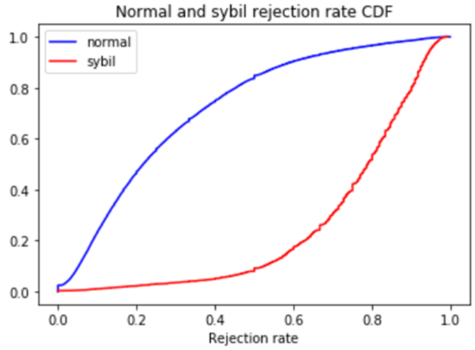


Fig. 4. The fraction of rejected requests of real users and Sybils.

We construct the PKU friend-request network as a signed graph with about 60K users and 2.2 million positive edges and 0.4 million negative edges. We use the 5.5K PKU fake accounts which have been detected by Renren security team as the ground truth of Sybils. We randomly choose 5% nodes from the ground truth as label data, and predict the labels of other 95% nodes. Here we choose a low labeling rate as labeled data is often expensive to obtain in practice, requiring manual examination of social network posts and user profiles.

Friend Request Graph. We construct the acceptance and rejections of friend request interactions of users as a signed graph $G = (V, E)$, where $V = \{v_1, ..., v_n\}$ represents n nodes and $E = \{e_{ij}\}$ represents the edges. Each edge e_{ij} is associated with a *sign* s_{ij}. indicating whether v_i accepts v_j ($s_{ij} = 1$) or v_i rejects v_j ($s_{ij} = -1$). The node feature information matrix is represented as $X = \{\boldsymbol{x_1}, \boldsymbol{x_2} ..., \boldsymbol{x_N}\}$ and $\boldsymbol{x_i} \in \mathbb{R}^d$. In our experiment, unique features are extracted from user-level activity [14], such as the frequency of friend requests, the fraction of accepted requests, the number of accepted or reject requests, clustering coefficient, etc.

Figure 2 and Figure 3 show the distribution of the number of accepted (rejected) requests of real users and Sybils, respectively. We see that real users have more accepted friend requests and less rejected friend requests than those of Sybils. Figure 4 shows the faction of rejected friend requests of real users and Sybils. On average, real users have a low rate of 0.3 to be rejected by others, whereas Sybils have a high rejection rate of 0.3. This is because real users typically send invitations to the people with whom they have prior relationships, whereas Sybils target strangers.

B. Attacking Strategies

We take the number of links already exist in the network as default network, and manipulate the links of Sybils to simulate more complex attacking scenarios, as illustrated in Fig. 5. We simulated the following attacking strategies in our experiment.

Collusion Attack. The attacker creates many positive fake edges among Sybils, e.g., Sybils send requests among themselves and accept each other.

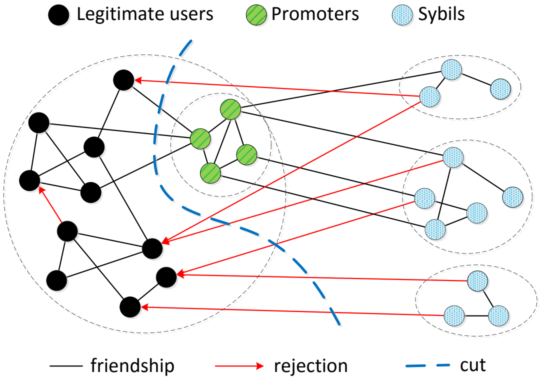


Fig. 5. Illustration of attack models.

Self-rejection Attack. The attacker attempts to whitewash a part of the Sybils by letting these Sybils reject friend requests from other Sybils.

Promotion Attack. Besides the manipulation between Sybils, Sybils could also send/receive requests to/from compromised real users who are more likely to accept. Here we suppose the number of compromised real users are quietly limited.

We set parameter α as the percentage of Sybils participating in the attack. The parameter β represents the percentage of collusion links, self rejection links or promotion links of these Sbyils in different attacking models, respectively. The parameter γ controls the probability that compromised users accept promotion requests.

C. GCN-based Detection Model

GCN Model. Given the graph, GCN uses a graph convolution operation to obtain node embeddings layer by layer. At each layer, the embedding of a node is obtained by gathering the embeddings of its neighbors, followed by one or a few layers of linear transformations and nonlinear activations. The final layer embedding is then used for some end tasks. Due to over-smoothing problem, a commonly used application is to apply two-layer GCN to semi-supervised node classification on graphs.

Figure 6 shows our model architecture. As the graph has two types of edges, we concatenate the features of node itself, the aggregated features of its positive neighbors and negative neighbors as the input for each layer. More formally, given the signed adjacency matrix A, let the positive matrix A_+ and negative matrix A_- contain only positive and negative values in the matrix A, respectively. In other words, $A = A_+ + A_-$. Let D_+ and D_- be the out-degree diagonal matrices: $D_{+ii} = \sum_j A_{+ij}$ and $D_{-ii} = \sum_j A_{-ij}$. Then the semi-row normalized matrix of A_+ (or A_-) is $\tilde{A_+} = D_+^{-1} A_+$ (or $\tilde{A_-} = D_-^{-1} A_-$).

In GCNs, each layer updates the node feature embedding in the graph by aggregating the features of neighboring nodes:

$$\begin{aligned} Z^{l+1} &= concat(X^l, \tilde{A}_+ X^l, \tilde{A}_- X^l) W, \\ X^{l+1} &= \sigma(Z^{l+1}) \end{aligned} \quad (1)$$

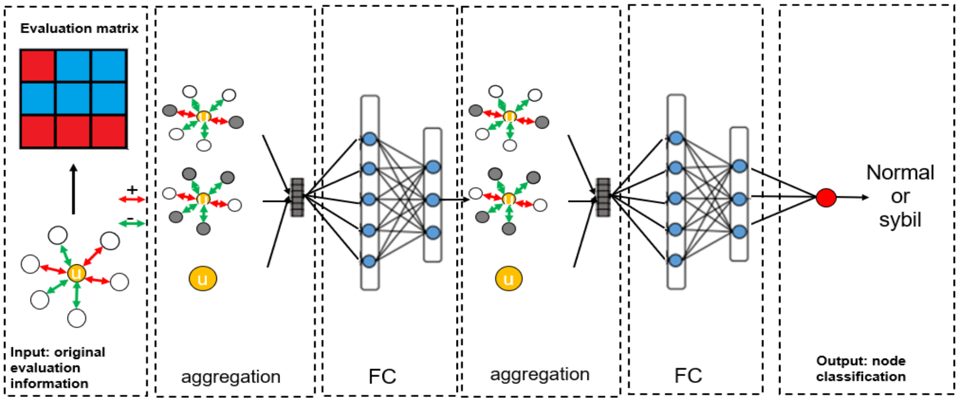
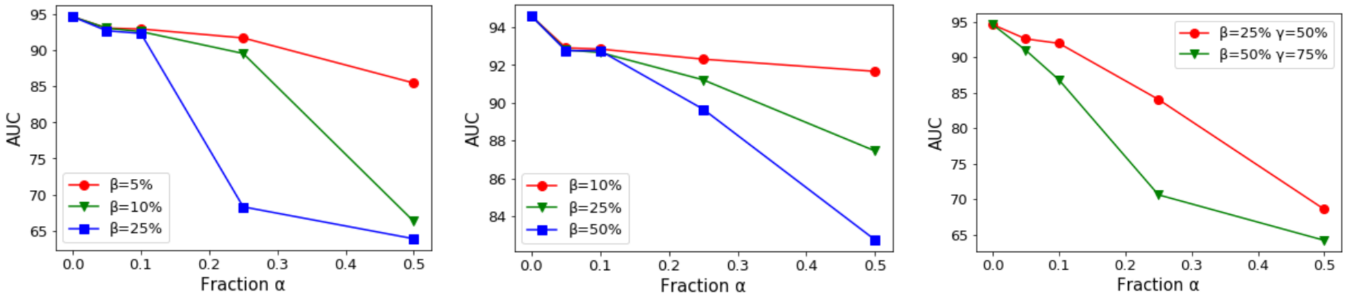

Evaluation matrix
+
−
u
Input: original evaluation information
aggregation
FC
aggregation
FC
Normal or sybil
Output: node classification

Fig. 6. The GCN-based Sybil detection model.



(a) Collusion attack

(b) Self-rejection attack

(c) Promotion attack

Fig. 7. Performance of GCN under different attacking strategies.

where X where X^l is the embedding at the layer l for all the nodes, and W is the feature transformation matrix which will be learnt for the node classification. The activation function σ is usually set to be the element-wise ReLU.

Model Setup. We set the hidden dimension of node as 16. In the training stage, we randomly initialize the model parameters with a Gaussian distribution, and optimize the model with mini-batch Adam. We set the batch size as 512, and set the learning rate as 0.01. All the experiments are implemented on the Tensorflow.

D. Detection Results

To our knowledge, we are the first to examine the robustness of GCN under different adverse manipulation in the social Sybil detection filed. Figure 7 shows the performance of GCN under different attacking strategies.

We see that the start point $\alpha = 0$ gives the performance of GCN before the attackers' manipulation. Even the available labeled data is limited (only 5%), GCN achieves high detection accuracy by capturing both the node-level and graph-level features. However, we see that the GCNs are vulnerable to adversarial manipulation on the local structure or feature of attacking nodes. All the proposed attacking strategy results in significant drops in performance for GCN as more Sybils participate in the manipulation.

For example, Figure 7(a) shows the resilience of GCN to the collusion among Sybils. We see that the performance of GCN drops dramatically as the number of collusion links grow, since Sybils could add fake positive links to make their computation graphs more similar to those of real users. We also observe that GCN is more vulnerable to the promotion attacks, where the accuracy drops significantly lower than others. Although the number of promoters (e.g., comprised real users) are quite limited, Sybils could enlarge their effect by allowing them to contribute the feature aggregation for each Sybil, making Sybils more difficult to distinguish from real ones.

The key reason behind high vulnerability is that the core idea of GCNs is to learn how to iteratively aggregate feature information from local graph neighborhoods using neural networks (Figure 1). Here a single "convolution" operation transforms and aggregates feature information from a nodes one-hop graph neighborhood. Thus, for a two-layer GCN, the attacker only needs to mimic the local structure and features of normal users by manipulating a small two-hop neighborhood by colluding with or self-rejecting other Sybils,

Moreover, we cannot solve this problem by stacking multiple such convolutions due to oversmoothing. Specifically, as the information can be propagated across far reaches of a graph, node representations become more and more similar, which eventually become indistinguishable. Also, it is still relatively easy for promotion attack to circumvent even given

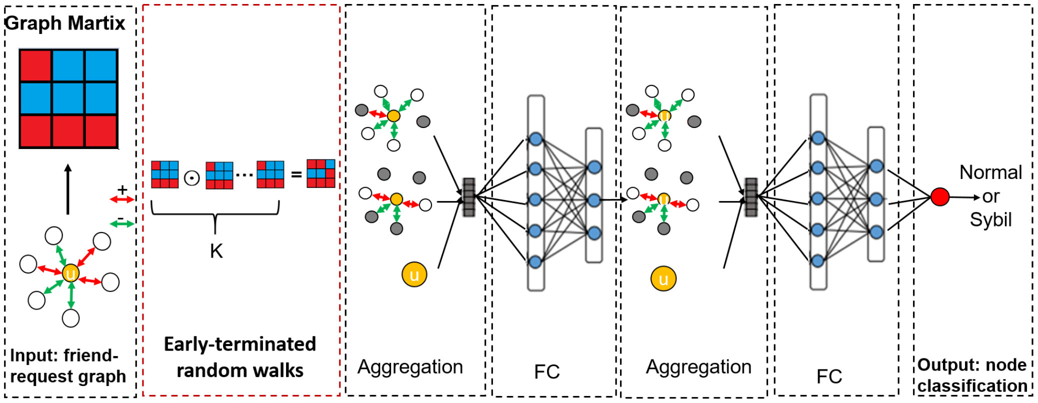

Graph Martix
+
-
u
Input: friend-request graph
K
Early-terminated random walks
Aggregation
FC
Aggregation
FC
Normal
or
Sybil
Output: node classification

Fig. 8. The workflow of TrustGCN model.

far reaches of a graph, because we are able to prevent from real nodes to be included for positive neighborhood aggregation once a promotion link is added. Therefore, the application of GCN on Sybil detection field brings a significant challenge to the robustness.

III. TrustGCN Model

To resolve the above major challenge, we propose the TrustGCN, which overcomes the limitations of GCN with social-graph-based methods.

A. Overview

Due to the difficulty of soliciting and maintaining reciprocal social relationships, we assume that Sybils have more rejected friend requests than accepted ones for signed friend request graph, or establish a limited number of attack edges to real users for unsigned social graph. The low fraction of accepted links (or limited attack edges) result in a negative cut (or a sparse cut) between the non-Sybil and the Sybil region.

To exploit the above structure gap, TrustGCN contains two key stages to achieve robustness while being able to introduce GCN into the Sybil detection filed, as shown in Figure 8:

Trust Propagation. TrustGCN first aims to ensure that most real accounts have higher trust than Sybils. It utilizes the power iteration method to efficiently propagate trust across the graph. This method involves successive matrix multiplications where each element of the matrix represents the random walk transition probability from one node to a neighbor node. Each iteration computes the landing probability distribution over all nodes as the random walk proceeds by one step.

Trust-guided Graph Convolution. TurstGCN then uses graph convolution operation (e.g., GCN) guided by the generated landing probabilities for node neighborhood aggregation and classification. Our goal is to ensure that the most features for graph convolution are collected from those trusted real users. Specifically, we introduce the landing probabilities as edge weights in the original graph to penalize the suspected fake links, thus is robust against attack manipulation on node classification.

We next describe each stage in detail. It is worth noting that TrustGCN could be used for both signed friend request graph or unsigned social graph. In the following, we mainly focus on the former, as the unsigned social graph could be treated it as a special case of containing no negative links.

B. Trust Propagation

Social-graph-based solutions uncover Sybils from the perspective of already known non-Sybil nodes (trust seeds). The intuition is that if we seed all trust in the non-Sybil region, then both trust/distrust can flow into the Sybil region via the both positive/negative links. If we terminate the power iteration early before it converges, non-Sybil users will obtain higher positive landing probability (trust), whereas the reverse holds for Sybils.

Signed Short Random Walks. We define $T_+^{(i)}(v)$ and $T_-^{(i)}(v)$ as the landing probabilities of signed random walks on node v after i iterations. Initially, $T-^{(0)}v = 0$, where as the total trust, denoted as $T_G (T_G > 0)$, is evenly distributed on a set of trust seeds S.

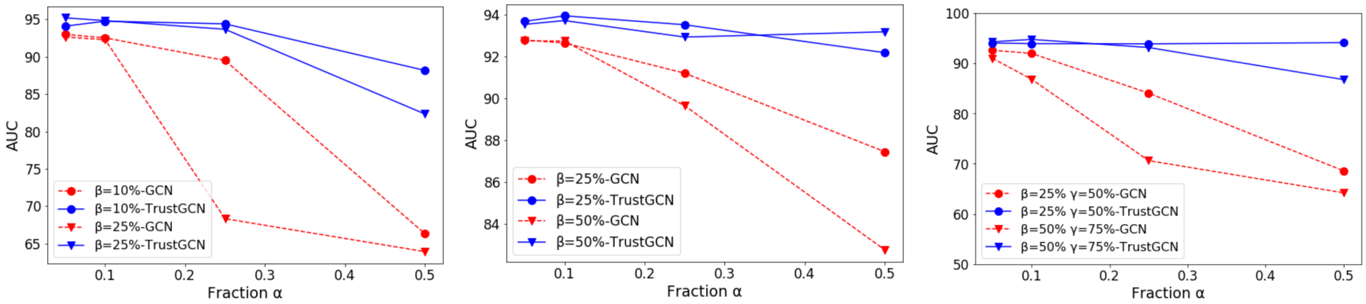
$$T+^{(0)}(v) = \begin{cases} \frac{T_G}{|S|} & v \in S \\ 0 & \text{else} \end{cases} \tag{2}$$

During each power iteration, a node first evenly distributes its trust/distrust to its neighbors according the sign of edges. It then collects trust distributed by its neighbors and updates its own signed landing probabilities accordingly. The process is shown below.

$$\begin{aligned} T+^{(i)}(v) &= \sum_{s_{uv}=1} \frac{T+^{(i-1)}(u)}{deg(u)} \\ T-^{(i)}(v) &= \sum_{s_{uv}=-1} \frac{T+^{(i-1)}(u)}{deg(u)} + \sum_{s_{uv}=\pm 1} \frac{T-^{(i-1)}(u)}{deg(u)} \end{aligned} \tag{3}$$

where $(u, v) \in E$.

If the random walk starting at node v with a positive sign encounters a negative edge, it flips the sign from positive to negative, or vice versa. Our model distinguishes whether node u is the friend of node v or not according to its sign at node u.



(a) Collusion attack

(b) Self-rejection attack

(c) Promotion attack

Fig. 9. The comparison between TrustGCN and GCN under different attacking strategies.

Different from structural balance theory, we still take enemy's enemy as enemy for detecting self-rejection attacks.

Early termination. We terminates the power iterations after small number of K iterations, e.g., O(log n) steps. This number of iterations is sufficient to reach a large portion over the fast-mixing non-Sybil region, but limits the trust escaping to the Sybil region or distrust flowing back to non-Sybil region.

C. Trust-guided Graph Convolution

Graph Weighting. Let T_+ and T_- be the positive and negative landing probability vectors for the nodes based on short random walks. TrustGCN incorporate these landing probability vectors into the graph as weights, such that edges incident to Sybils have lower weights than others for the feature aggregation. More formally, let A be the signed adjacency matrix, we weigh the graph as follows:

$$\begin{aligned} T &= T_+ + T_-, \\ A^{tr} &= TA \end{aligned} \tag{4}$$

Graph Convolution. Intuitively, a large edge weight implies the source node is more trustable and reliable for the feature aggregation of the target node. Thus, we perform graph convolution operation on this new weighted graph defined by weighted adjacent matrix A^{tr}. Let the positive matrix A^{tr}_+ and negative matrix A^{tr}_- contain only positive and negative values in the matrix A, respectively. Let D^{tr}_+ and D^{tr}_- be the weighted out-degree diagonal matrices: $D^{tr}_{+ii} = \sum_j A^{tr}_{+ij}$ and $D^{tr}_{-ii} = \sum_j A^{tr}_{-ij}$. Then the semi-row normalized matrices of A^{tr}_+ and A^{tr}_-) are $\tilde{A}^{tr}_+ = (D^{tr}_+)^{-1} A_+$ and $\tilde{A}^{tr}_- = (D^{tr}_-)^{-1} A_-$, respectively.

Based on the above definitions, a single layer of TrustGCN model can be represented as the following equation:

$$\begin{aligned} Z^{l+1} &= concat(X^l, \tilde{A}^{tr}_+ X^l, \tilde{A}^{tr}_- X^l)W, \\ X^{l+1} &= \sigma(Z^{l+1}) \end{aligned} \tag{5}$$

where X^l is the embedding at the layer l for all the nodes, and W is the feature transformation matrix which will be learnt for the node classification.

Since the graph is weighted by the trust score, TrustGCN could achieve more reliable feature aggregation, thus is robust against adverse manipulation over the signed friend request graph. For example, most the collusion links between Sybils would have negative weights given the negative cut between the Sybil and normal regions. As a result, the Sybil partners contribute each negative neighborhood feature aggregation instead of positive ones in the original graph, thus making them more easy to be distinguished from real users.

TrustGCN is also robust against adverse manipulation over the unsigned social graph. Given the assumption that OSN Sybils have a disproportionately small number of connections to real users, the contribution of Sybils to the feature aggregation of each other is limited by this small number of attack edges, irrespective of their manipulation.

IV. Results on Robust Detection

We use the same experiment set up as Section 2. For TrustGCN, we choose the propagation iteration number K as 4, with other parameters similar to the GCN used in Section 2. Figure 9 shows the performance comparison between TrustGCN and GCN under different attacking strategies. We make the following observations.

First, we see that TrustGCN significantly enhances resilience to each types of attacks compared to GCN. The fundamental rationale of TrustGCN is to leverage the unique structural gap of Sybil community in this signed graph: the colluding Sybils as a whole has more negative outgoing links than positive ones, since real users usually send/accept the friend requests to/from their friends or acquaintances. Although an attacker can control the connections between Sybils arbitrarily, it is hard to manipulate such structural gap between benign nodes and Sybils. Because of high fraction of negative landing probability on himself or other Sybil neighbors, the Sybils cannot involve their partners for feature aggregation, irrespective how they manipulate local neighborhood.

Second, TrustGCN is much robust in the self-rejection and promotion attacks. Since we still consider enemy's enemy as enemy, the attacker cannot whitewash a part of the Sybils by adding self-rejection links among Sybils. For the promotion attack, the Sybils could get some positive landing probability from compromised or popular users that are more willing to accept other strangers. However, this probability is quite small as these users have a large number outgoing edges but with fixed incoming edges and trust scores. This prevents Sybils from positively contributing to the feature aggregation of each other due to low trust scores (i.e., the low weights in feature aggregation).

Finally, we observe that TrustGCN has slight accuracy drop with the increasing number of positive collusion links. This is because some inactive Sybils have not launched attacks yet. As they receive few negative links whereas many positive links under the collusion strategy. The model tends to mis-classify them as normal users. These accounts cannot do harm to real users until after they start generating friend requests to real users or their colluders. In this case, TrustGCN can recall these accounts as quickly as possible once they become active, to minimize the amount of damage they can do to real users.

V. Related Works

Sybil attacks are among the fundamental threats for distributed systems, especially for todays online social networks.

Feature-based detection. This approach relies on user-level malicious behaviors manifested as profile and activity features, and use various machine learning techniques to classify each account as fake or real. Yang et al. [14] use activity-level features, such as frequency of friend requests, fraction of accepted requests, and per-account clustering coefficientus to train an SVM classifier in order to detect fake accounts. Íntegro [15] employs feature-based detection to identify unknown victims in a non-adversarial setting. The dataset used to train a victim classifier includes features of only known real accounts that have either accepted or rejected friend requests send by known fakes. Even though feature-based detection scales to large OSNs, it is still relatively easy to circumvent, since attackers can cheaply create fakes that resemble real users.

Social-graph-based defenses. Yu et al develop SybilGuard [7] and SybilLimit [16] to defend against Sybil attacks, which are among the first approaches leveraging the structure of social networks. These two systems rely on the basic idea that normal users and Sybil nodes are respectively well connected and mutually isolated. Thus random walk from normal users can hardly reach Sybil users, which could help to classify the community of normal users and Sybils. These schemes encourage a thread of research based on the edge conductance [8], [10], [17]–[20]. They leverage the random walk method and additional structural characteristics of the social network to distinguish the Sybil communities from the normal users. The primal prerequisite to use these random walk based mechanisms is that the communities should be externally isolated and internally fast mixing.

Another direction is to convert the Sybil identification problem to community detection problem. Work from [21] and [22] show the performance of leveraging community detection techniques to achieve Sybil detection job. The community detection algorithms in [23] and [24] offer methods to transfer the Sybil detection problems into the community detection problems. Stemming from the observation that Sybils inevitably receive a significant number of social rejections from legitimate users, previous studies [13], [25] begin to take into account request rejections.

However, graph-based defenses cannot leverage the feature information for high classification accuracy. In practice, they are usually used to derive a quality ranking instead of binary classification, in which a substantial portion of Sybils ranks low. This enables the OSN to focus its manual inspection efforts towards the end of the list, where it is more likely to encounter Sybils.

Graph Neural Networks. Recent years have seen significant developments in this space especially the development of new deep learning methods that are capable of learning on graph-structured data. Most prominent among these recent advancements is the success of deep learning architectures known as Graph Convolutional Networks (GCNs) [11], [12], [26], [27]. These methods combine standard neural networks with iterative graph propagation: the feature of a target node is computed recursively (with neural networks) from features of graph-structural neighborhood nodes.

However, despite the successes of GCN algorithms, no previous works have managed to apply them to Sbyil detection field in a Sybil-resilient manner. Previous works [28], [29] develop algorithms to attack GCNs by perturbing the links and embeddings of a small subset of nodes. They employed a gradient ascent method to change the graph structure in the whitebox setting. Our methods focus on Sybil detection scenario in social network and have different attacking strategy and proposed defenses from the previous works.

VI. Conclusion

In this paper, we presented TrustGCN, a robust Sybil detection algorithm that combines traditional social-graph-based defenses with recent advancements in graph neural networks. TrustGCN achieves this combination with a two-stage framework: It first computes trust scores of nodes in the graph based on the land probabilities of short random walks starting from known benign nodes. Given the structure gap between Sybils and real users, most Sybils would have significantly lower trust scores than those of real users. Then TrustGCN integrates these trust scores of nodes into the graph convolution operation as edge weights (or the probability for selection), in order to guarantee that most neighbors for feature aggregation are selected from those real users. Using the proposed approach, we are able to leverage the advantage of GCN to incorporate both graph structure as well as node feature information for more accurate Sybil detection, while significant improving the GCN's robustness against adversarial attacks and manipulation over the graph.

Acknowledgment

Authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by the National Natural Science Foundation of China under Grant No. 61972004.

References

[1] J. Douceur, “The sybil attack,” *Peer-to-Peer Systems*, pp. 251–260, 2002.

[2] Q. Lian, Z. Zhang, M. Yang, B. Y. Zhao, Y. Dai, and X. Li, “An empirical study of collusion behavior in the maze p2p file-sharing system,” in *27th International Conference on Distributed Computing Systems (ICDCS ’07)*. IEEE, 2007, pp. 56–56.

[3] G. Noh, H. Oh, K.-h. Lee, and C.-k. Kim, “Toward trustworthy social network services: A robust design of recommender systems,” *Journal of Communications and Networks*, vol. 17, no. 2, pp. 145–156, 2015.

[4] G. Noh and H. Oh, “Influence level-based sybil attack resistant recommender systems,” in *2014 IEEE Fourth International Conference on Big Data and Cloud Computing (BdCloud)*. IEEE, 2014, pp. 524–531.

[5] K. Bauer, D. McCoy, D. Grunwald, T. Kohno, and D. Sicker, “Low-resource routing attacks against tor,” in *Proceedings of the 2007 ACM workshop on Privacy in electronic society*. ACM, 2007, pp. 11–20.

[6] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao, “Detecting and characterizing social spam campaigns,” in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010, pp. 35–47.

[7] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, “Sybilguard: Defending against sybil attacks via social networks,” in *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM ’06)*. ACM, 2006, pp. 267–278.

[8] G. Danezis and P. Mittal, “Sybilinfer: Detecting sybil nodes using social networks.” in *Proc. of NDSS*. San Diego, CA, 2009.

[9] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, “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 (NSDI ’12)*. USENIX Association, 2012, pp. 15–15.

[10] N. Z. Gong, M. Frank, and P. Mittal, “Sybilbelief: A semi-supervised learning approach for structure-based sybil detection,” *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 976–987, 2014.

[11] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *International Conference on Learning Representations*, ser. ICLR’17, 2017.

[12] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, “Graph neural networks: A review of methods and applications,” *arXiv preprint arXiv:1812.08434*, 2018.

[13] 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 *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM ’13)*. IEEE, 2013, pp. 2400–2408.

[14] 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.

[15] Y. Boshmaf, M. Ripeanu, and K. Beznosov, “Íntegro: Leveraging victim prediction for robust fake account detection in osns,” in *Proc. of NDSS*, 2015.

[16] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, “Sybillimit: A near-optimal social network defense against sybil attacks,” in *IEEE Symposium on Security and Privacy (SP ’08)*. IEEE, 2008, pp. 3–17.

[17] D. N. Tran, B. Min, J. Li, and L. Subramanian, “Sybil-resilient online content voting.” in *Proc. of NSDI*, vol. 9, no. 1, 2009, pp. 15–28.

[18] N. Tran, J. Li, L. Subramanian, and S. S. Chow, “Optimal sybil-resilient node admission control,” in *Proc. of INFOCOM*. IEEE, 2011, pp. 3218–3226.

[19] W. Wei, F. Xu, C. C. Tan, and Q. Li, “Sybildefender: Defend against sybil attacks in large social networks,” in *Proc. of INFOCOM*. IEEE, 2012, pp. 1951–1959.

[20] T. G. Kolda and M. J. Procopio, “Generalized badrank with graduated trust,” *Sandia National Laboratories, California*, 2009.

[21] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove, “An analysis of social network-based sybil defenses,” *ACM Special Interest Group on Data Communication*, 2010.

[22] L. Alvisi, A. Clement, A. Epasto, S. Lattanzi, and A. Panconesi, “Sok: The evolution of sybil defense via social networks,” in *2013 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2013, pp. 382–396.

[23] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, “You are who you know: inferring user profiles in online social networks.” in *Proceedings of the 3rd ACM International Conference of Web Search and Data Mining (WSDM ’12)*, 2010, pp. 251–260.

[24] Z. Cai and C. Jermaine, “The latent community model for detecting sybils in social networks,” in *Proc. of NDSS*, 2012.

[25] Q. Cao, M. Sirivianos, X. Yang, and K. Munagala, “Combating friend spam using social rejections,” in *Proceedings of the 35th IEEE International Conference on Distributed Computing Systems (ICDCS ’15)*. IEEE, 2015.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” in *Advances in Neural Information Processing Systems*, ser. NIPS’16, 2016, pp. 3844–3852.

[27] W. L. Hamilton, R. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in *Advances in neural information processing systems*, ser. NIPS’17, 2017, pp. 1024–1034.

[28] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song, “Adversarial attack on graph structured data,” in *35th International Conference on Machine Learning*, ser. PMLR’18, 2018.

[29] D. Zugner, A. Akbarnejad, and S.Gunnemann, “Adversarial attacks on neural networks for graph data,” in *ArXiv e-prints*, 2018.