Follow Spam Detection based on Cascaded Social Information

Sihyun Jeong[a], Giseop Noh[a], Hayoung Oh[b], Chong-kwon Kim[a,*]

[a]Dept. of Computer Science and Engineering, Seoul National University, Gwanak-gu, Seoul 151-744, Republic of Korea
[b]School of Electronic and Engineering, Soongsil University, Dongjak-gu, Seoul 156-743, Republic of Korea

Abstract

In the last decade we have witnessed the explosive growth of online social networking services (SNSs) such as Facebook, Twitter, RenRen and LinkedIn. While SNSs provide diverse benefits for example, forstering inter-personal relationships, community formations and news propagation, they also attracted uninvited nuiance. Spammers abuse SNSs as vehicles to spread spams rapidly and widely. Spams, unsolicited or inappropriate messages, significantly impair the credibility and reliability of services. Therefore, detecting spammers has become an urgent and critical issue in SNSs. This paper deals with Follow spam in Twitter. Instead of spreading annoying messages to the public, a spammer follows (subscribes to) legitimate users, and followed a legitimate user. Based on the assumption that the online relationships of spammers are different from those of legitimate users, we proposed classification schemes that detect follow spammers. Particularly, we focused on cascaded social relations and devised two schemes, TSP-Filtering and SS-Filtering, each of which utilizes Triad Significance Profile (TSP) and Social status (SS) in a two-hop subnetwork centered at each other. We also propose an emsemble technique, Cascaded-Filtering, that combine both TSP and SS properties. Our experiments on real Twitter datasets demonstrated that the proposed three approaches are very practical. The proposed schemes are scalable because instead of analyzing the whole network, they inspect user-centered two hop social networks. Our performance study showed that proposed methods yield significantly better performance than prior scheme in terms of true positives and false positives.

Keywords: Online Social Network, Security, Twitter, Follow Spam, Triad, Status theory

*Corresponding author
Email addresses: sihyunj@snu.ac.kr (Sihyun Jeong), gsno@popeye.snu.ac.kr (Giseop Noh), hyoh@ssu.ac.kr (Hayoung Oh), ckim@snu.ac.kr (Chong-kwon Kim)

1. Introduction

The use of social networking services (SNSs) continues to grow exponentially with the widespread adoption of smart devices such as smart phones, smart pads, smart watches, and so on. SNSs can connect people and can be used to share information in real time. SNSs such as Facebook, Twitter, and Ren-Ren are becoming the most influential mediums for building social relations, as well as for the sharing and propagation of information. According to recent announcement, Twitter, one of the largest and the most popular SNSs, passed 255m monthly active users and expects 80% of advertising revenue from mobile users[1]

After repeated explosive growth in user population, matured SNSs such as Facebook and Twitter become a necessings in modern life in developed countries. In addition, relatively new SNSs such as RenRen and Sina Weibo, targeted for specific country or language speakers, replicate the eruptive expansion of the earlier SNSs. For example, an influential user can be exploited by a person working in online marketing to maximize the marketing effect; malicious users (attackers) disseminate false information or fraudulent messages for the purpose of phishing, scam, or malware intrusion. That is, the attackers post multiple unrelated messages with trending topics to attract legitimate users and encourage them to click the malicious links in the messages.

Spam refers to unwanted messages from unknown sources (attackers). One of the major negative aspects of SNS is spam. In the early Internet era, spam appeared in emails or SMS (short message service). However, the domain of spam expanded into SNS as the popularity and usage of the services continued to increase. False information from SNS can spread rapidly in real time. *Follow spam* was reported recently and is a system that tries to increase the number of relations (or friendships) in users networks for the purpose of sending spam via SNS. The attack pattern of the follow spam begins with the attacker disseminating spammer accounts that follow a large number of legitimate users for the purpose of receiving a follow-back or drawing attention to the spam account [15]. Due to the consequent exposure of the public to spam content, this practice definitely lowers the reliability of SNS.

In practice, Twitter has experienced Follow spam problems, reducing users trust in message distribution and increasing computation overhead. In 2008, Twitter officially announced that Follow spam accounts had followed so many people that they threatened the performance of the entire system [2]. Even with the emerging threat from Follow spam, it has been barely investigated or researched. A contents-based spam filtering approach is employed in the Twitter spam field [12, 5, 28, 42]. However, since spam contents keep changing to avoid content-based detection by inserting URLs and images in spam messages, the contents-based approach is vulnerable against evolving message patterns. To

---

[1]http://thenextweb.com/twitter/2014/04/29/twitter-passes-255m-monthly-active-users-198m-mobile-users-sees-80-advertising-revenue-mobile/

[2]https://blog.twitter.com/2008/making-progress-on-spam

overcome the limitations of the content-based approach, a new approach using inherent properties of SNS was introduced.

[15] first emphasized that *Follow spam* should be detected by using its link-farming property. They proposed a PageRank-based ranking algorithm to lower the impact of spammers. However, this approach can be burdensome since it needs to utilize social network data for the entire network (i.e., all information for nodes and edges). Therefore, it has a high computational cost and can barely detect *Follow spammers* in real time. As a result, a novel detection mechanism with low computational cost and real time spam filtering is needed while maintaining the detection performance. In this paper, we suggest two social network-based detection schemes for countering Twitter spam. First, spammer accounts are filtered out with the use of a Triad Significance Profile (TSP) that measures the structural differences between the frequencies of 13 isomorphic subgraphs. We discovered that TSP of a spammer account is different from that of a legitimate users account with only 2-hop social networks. According to our experiment, 92.1% of spammers are classified correctly when we used only TSP features for classification. This result suggests that frequency and distribution of isomorphic subgraphs could be informative features for identifying spammers. Secondly, we introduce a new detection approach using the social status (SS) theory [27] to distinguish spammer accounts. Legitimate users typically follow accounts of a higher status than themselves, whereas spammers are likely to follow in a random manner. With these approaches, we can confirm that cascaded social network-based approaches (TSP and SS) can effectively detect *Follow spammers* with a low cost.

Our experiments on real Twitter datasets clearly show that our three mechanisms, TSP-Filtering, SS-Filtering and Cascaded-Filtering, are very practical for the following reasons. First, our approaches require only a small user-related 2-hop neighborhood social network. Actually, there are only few existing works focused on small neighborhood graph in other areas [30, 1], but none of them discovered the power of neighborhood social network clearly. Therefore, they can be applied to spam detection systems in social networks as real time solutions. Second, service providers can maintain the credibility and reliability of their SNSs by applying our approaches. Legitimate users are less likely to be blocked by the system with low false positives (5.7%). Also, a high proportion of true positives (96.3%) provides a secure environment to users.

Our main contributions are summarized as follows:

- For the first time, we discovered the feasibility of cascaded social information such as triad (or isomorphic subgraph) and social status based positive link probability as good features for classifying spammers and legitimate users in Twitter;

- Our approaches involve more lightweight computation for real time spam detection than the previous scheme (i.e., global information). Since to check whether a certain user is spammer or not, we only focused on the 2-hop social networks of each user (i.e., local information). and extract cascaded social information from the network;

3

- Based on Twitter's spam policy, novel triad graph based features and social status theory based features are proposed and cascaded together to facilitate spam detection;

- To the best of our knowledge, our approaches are the first experiments with real world data to provide the credible and reliable Twitter system with true positive results of up to 96.3%. We believe that our findings can provide valuable insights to the area of spam detection and defense in various social networks;

- To sum up, this paper is the first work which clarify the difference between spammer and legitimate user in the view of subgraph consisting of neighborhood. Moreover, we suggest the novel three approaches such as TSP-Filtering, SS-Filtering and Cascaded-Filtering for the follow spam detection based on cascaded social information.

The rest of the paper is organized as follows: Firstly, we introduce interesting related works on twitter spam and spam detection mechanisms in section 2. Then, to describe our approach, we introduce the motivation for our work in section 3. Then, we propose the TSP-Filtering and SS-Filtering methods with the respective performance evaluation in section 4 and section 5. In Section 6, we propose Cascaded-Filtering with higher true positive and lower false positive than TSP-Filtering and SS-Filtering. Section 7 presents an overall evaluation and discussion of our three approaches (TSP-Filtering, SS-Filtering and Cascaded-Filtering) and Collusionrank [15]. Lastly, we close this paper with future work and the conclusion in section 8 and 9.

2. Related work

2.1. Twitter-spam Filtering

2.1.1. Content based Twitter-spam Filtering

Twitter contents such as user profile, tweets, and the activity log provide various options for distinguishing spammers from legitimate users. Spammers generally write tweets that contain a hashtag and URL according to the following research studies that analyzed commonly used hashtags and URL: [12, 5, 28, 42].

COMPA [12] detected compromised accounts that wrote spam tweets based on the tweeting language of the user's account, the tweeting time window, the URL, and the "mention" receiver. This is a personalized detection approach that learns the previous behavioral pattern of each user. Benevenuto et al. [5] and Martinez-Romo et al. [28] proposed classification models that learned the number of hashtags and URLs [5] or spam URLs that are used in spam groundtruth tweets. Yardi et al. [42] studied spammers' strategic behavioral patterns and also concluded that the use of hashtags related to trending topics is a very effective spamming strategy. Gao et al. [14] built a template based on the sentence structure of spam groundtruth tweets and used template matching to filter out spam tweets.

4

2.1.2. Social-network based Twitter-spam Filtering

As the attack strategies of Twitter spammers have become more effective, a variety of social-network-based Twitter-spam-detection approaches have been introduced.

Twitter spammers attempt to convince the public that their accounts are legitimate or famous, and they display their spam tweets to the public. It is easier for spammers to attract user interest by exploiting as much social information as possible. Previous works identified follower-market customers who purchased fake accounts that they used to follow themselves [22, 34]. Jiang et al. [22] analyzed the behavioral synchronicity among these fake accounts to detect their presence. Stringhini et al. [34] examined real follower markets and detected fake accounts by identifying those for which the number of followees increased suddenly but did not decrease any further. Viswanath et al. [36] used Principal component analysis (PCA), an anomaly-detection approach, to detect intentional "follow" or "like" behaviors from market customers.

On Twitter, *Follow spam* (or link farming) refers to the act of following a mass number of people to garner attention or follow-backs[3]. Ghosh et al. [15] pointed out that most *Follow spammers* attained a higher rank in existing ranking algorithms because their reciprocal-follower rate is 82%. Based on their findings, they proposed the application of Collusionrank.

2.1.3. Subnetwork based Spam Filtering

The authors of [37] directly crawled Twitter's data and analyzed them with both contents and social graph modeling based approaches. Based on analysis of the contents, categorized into legitimates and spams, they proved that their proposed reputation feature has the best performance among all social graph-based features for detecting abnormal behaviors. However, they only considered the relationship between outdegrees and indegrees in a simple Twitter graph for the proposed reputation feature. Even though this scheme also utilizes a small graph (subgraph), a sophisticated graph design is as only part of the triad approach. The authors of [30] used neighborhood subnetwork (i.e., ego network) to detect comment spammers in Youtube. They also utilized selected discriminating motifs and analyzed them in Youtube video-user relation network. It seems very similar with our work, but it used spam campaign related motifs. Therefore, it cannot distinguish spammers when they use other sophisticated strategy. [1] extracted weighted subgraphs from target network and utilizes them as discriminating features to detect spammers. It also analyzed subgraphs by types of anomalies. Based on power law characteristic of social network, it compared spammer to legitimate users neighborhood subnetwork in terms of edge or weight distribution.

---

[3]https://blog.twitter.com/2008/making-progress-spam/

2.2. Link spam Filtering

Link spam has been widely studied in the web spam detection field. This type of spam is presented as numerous links from a large number of web pages to a few target web pages. Studies on *Link spam* have been receiving attention due to the limitations of PageRank [31] and HITS [24]. Thanks to significant link characteristics, many web link graph structure-based spam detection approaches have been introduced [18, 40, 25, 3, 41, 8]. TrustRank [18] is one of the most popular *Link spam* detection algorithms. It propagates the 'non-spam' label through social networks. Likewise, BadRank [40] propagates the 'spam' label through social networks. Compared to PageRank [31], these two algorithms utilize 'non-spam' and 'spam' label propagation to lower the rank of spam webpages. [4] proposed an advanced *Link spam* detection algorithm using both 'spam' and 'non-spam' label propagation. These label propagation algorithms require seed knowledge such as a set of spam nodes and a set of non-spam nodes. Therefore, noise in the initial dataset can be a critical issue for these algorithms.

2.3. Sybil Detection

Most SNS spam detection systems rely on Sybil detection algorithms. Peer-to-peer systems consist of multiple nodes with several connections (edges). The system has to ensure that each node is clearly identified; otherwise, a malicious user (Sybil) can attempt to create multiple fake identities masquerading as honest nodes [11]. They can then manipulate the system (by zombie machines) or attack the system in order to gain illegal profit such as positive feedback in the reputation system, getting more votes in internet polls, or targeting sites to increase their rank in Google PageRank.

There are two main approaches to the Sybil attack: centralized and decentralized. Centralized defense obtains admission control through a central authority. Decentralized defense has no trusted central authority and controls the IP address by binding an identity. For the decentralized attack, SybilGuard [44] proposes that when each node receives $\sqrt{k}$ independent samples from a set of honest nodes of size k, a random walk can be performed to try to discover the Sybil identities by using the intersection probability between honest and Sybil groups. The number of available attack edges in Sybil are theoretically bound $O(\sqrt{k}\log k)$. SybilLimit [43] is an enhanced method introduced by [44]. They reduced the attack edge bound in near optimal $O(\log k)$ by exploiting various random walk methods. GateKeeper [35] adapts the ticket distribution algorithm to obtain each node's probability of Sybil/honest users.

Secondly, the centralized method. SybilInfer [10] assumed that the central authority knows the entire social network. After random walks, each node is assigned a Sybil/honest probability by measuring the Bayesian inference. SybilDefender [39] assumed that when starting a random walk in Sybil nodes, it will pass the intersection between honest and Sybil nodes. These approaches apply community detection algorithms to find Sybil communities. SumUp [35] addresses the vote aggregation problem by considering each voter's trust graph and calculating a set of max-flow paths from all voters.

Currently, there are many Sybil detecting methods with various social network properties. SybilRank [6] investigates each node by assuming that honest nodes will have higher degree-normalized landing probability. A random walk is performed to measure the ranking to determine whether the account is Sybil or not. SybilShield [33] utilizes a multi-community social network structure environment, considering sociological properties to cut the edge between honest and Sybil groups, performing modified random walks and figuring out the properties of multi-hop edges. SybilBelief [16] detects Sybil nodes based on a semi-supervised learning framework. This method modifies the Loopy Belief propagation system and the pairwise Markov random field to define each node's classification (Sybil/honest).

2.4. Data Mining Approach for spam detection

In spam detection problem, most of existing studies related the problem to classification task as follows. In general, spam classifier firstly learn features extracted from SNS using pattern of legitimate users such as the number of followees/followers, post uploading time and contents information of user profile and posts. Then, classifier determines if newly given test user is spammer or legitimate user by comparing to learned pattern. Therefore, if the test users behavioral pattern feature is far from legitimate users pattern feature (learned feature), the classifier could classify and detect the user as spammer. In some cases, classifiers adopt classification threshold to handle the tradeoff between true positive and false positive. Since reliability and credibility is crucial in using SNS, low false positive is treated particularly according to spam detection system.

In detail, [23] used linear regression for classifying and detecting spammers and it stated that deviant users from legitimate users patterns could be classified as spammers. Similarly, [36] utilized PCA (Principal Component Analysis) and it detected Facebook spammers who are distant from the principal component of legitimate users. Also, Markov random field based spam classification approach was proposed in [13]. Especially, contents-based spam detection approaches largely used Nave bayes classifier or SVM classifier with contents-related features. In the early stage of spam detection, [17, 21] and many similar studies analyzed token or word in spam contents and applied extracted features to the Nave bayes classifier. [32] proposed optimized version of SVM spam classifier and achieved efficiency than previous ones. [45] relieved false positive problem by adopting boundary region to classification result. Since most of spam classification is binary classification of spam and non-spam, ternary classification gives three classification labels including boundary region which means reconsidering region.

3. Motivation

3.1. Web, Social Network and Twitter

Like the Web, where the importance of each page is largely determined by who references whom, the influence of individuals on many SNSs is determined

by the number of indexes they receive. For example, the number of followers is the most important factor on Twitter and determines social capital, while the number of "likes" on Facebook is similar. This feature, however, has attracted a plethora of frauds who try to increase the importance or reputation of entities by generating bogus indexes, leading to the definition of the spamdexing class of attacks. Twitter's size has expanded exponentially over the past several years and it now has over 255 million active users after a succession of rapid growth spurts that resulted in an average annual growth rate of 25% [4]. Notably, the social-interaction structure of Twitter is very interesting. Users can follow famous persons–usually celebrities or standout opinion leaders–that they are unacquainted with, as well as close friends. Therefore, Twitter plays an information-propagation role in addition to the role of an online social network [26].

More importantly, contrary to the Web, Facebook, and many other social networks where spam indexes usually originate from fake accounts and from a circle of colluding link farms, a malicious person can collect followers or fans from innocent, social-capital-conscious users on Twitter. Further, the high rate of follow-backs makes the detection of Twitter spammers more difficult because they receive many legitimate followers just by following target users [15]. Existing link-farm-detection methods well fitted for web spam detection field therefore lose much of their effectiveness in the detection of spamdexing on Twitter.

In this paper, we demonstrate the feasibility of a cascaded SNS-based security scheme to detect *Follow spam*. Different from the unpractical and heuristic approaches of previous works, with the characteristics of follow-backs we apply triad frequencies and status theory for the first time in our *Follow spam* detection scheme. Note that the main purpose of this study is not the attainment of engineering optimization for the performance enhancement of prior schemes, but rather, it is the examination of the feasibility of a social-network-based security scheme in a popular online social networking site, i.e. Twitter.

Before we formalize the problem, we address the characteristics of Twitter. All 13 types of directed social graph models and social status with local information can be observed in Twitter. Additionally, Twitter has well-defined social relations in the form of the "follower" and "friend" relationships. In addition to these characteristics, spams show up frequently in Twitter. We practically exploit the policy of Twitter against spams to design our proposed scheme. Twitter's spam policy is summarized as follows:

- "If you have a small number of followers compared to the amount of people you are following", the account may be considered a spam account.

- "Multiple duplicate updates on one account" is a factor used to detect spam.

---

[4]http://thenextweb.com/twitter/2014/04/29/twitter-passes-255m-monthly-active-users-198m-mobile-users-sees-80-advertising-revenue-mobile/

- "If your updates consist mainly of links, and not personal updates", it is considered spam.

First policy is related with the social-interaction structure of Twitter while second and third policies have to do with spam contents. Most previous works focused on contents analysis or full information usage of social networks with a high amount of computational overhead. Different from previous approaches considering second and third policies, we accurately detect *Follow spam* using only local information of the social-interaction structure of Twitter. That is, our cascaded social network scheme is applicable regardless of the content such as Tweet, time and links.

3.2. Link spam and Follow spam

The concept of link farming originated from Web spam. The intent of link and *Follow spam* is to increase the population of a specific (target) website or reputation. Since normal search engines (e.g., Google) place popular websites on the first page, link-farming websites create numerous links to the target website.

PageRank [31], the most popular website ranking algorithm, ranks websites based on the indegree of the site. Actually, the popularity of inlink nodes is also important, but numerous inlinks are likely to increase the target website's ranking. Therefore, link farms generally contain plural links, and the links are created from many nodes to a few target nodes.

Follow spam, a special attack strategy on Twitter, has been shown to be a link farming technique. Figure 1 shows an example of *Follow spam*.
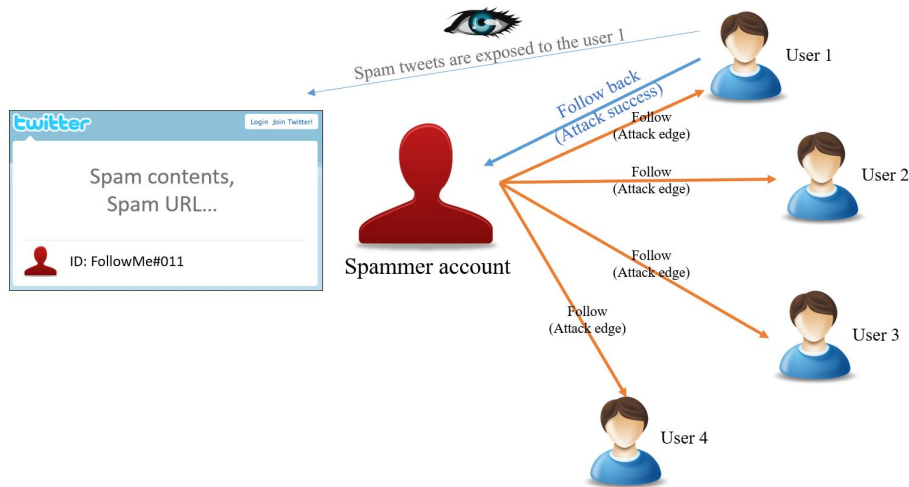

Figure 1: Overview of *Follow spam*

Table 1: Twitter dataset

The number of total users	The number of spammers
54,981,152	41,352

Follow spam consists of numerous links, but some differences exist. First, links are created by a few spammer nodes and they target many legitimate user nodes. More specifically, original link spam denotes many spammer nodes-few legitimate nodes relationship while *Follow spam* denotes few spammer nodes-many legitimate nodes. Second, the purpose of *Follow spam* is not just linking, but receiving a follow-back (reciprocal link). A user on Twitter can see tweets (contents) from another user when he/she follows (subscribe) the other's account. Consequently, spammers need to be followed by other users to show their spamming contents such as URL, image and advertisement.

Therefore, to gain more followers and attention, spammers send a large number of follow links to legitimate users. Surprisingly, the majority of followers who *Follow spam* accounts have been previously targeted by spam accounts. To be specific, 82% of legitimate users send a follow-back to spammers [15]. If s is a spammer account and his/her outlinks are all attack edges for follow back, the attack strength of s ($AS(s)$) is defined in (1). We defined the ratio between successful follow spam links (follow back links) of spammer s ($N_{fb}(s)$) and total follow spam links of s ($N_f(s)$) as $AS(s)$ as follows :

$$AS(s) = N_{fb}(s)/N_f(s) \tag{1}$$

In (1), $N_f(s)$ has the same meaning of outdegree of s. Therefore, the attack strength ($AS(s)$) of follow spam relies on a successful number of follow backs.

3.3. Twitter Dataset and Collusionrank

We conducted an experiment with a large-scale Twitter-follow link dataset that was provided by MPI-SWS [9]. This dataset was collected in September 2009, contains 1,963,263,821 directed social links, and the number of corresponding users is 54,981,152. We also used the *Follow spammer* dataset from [15] that contains 41,352 spammers as the ground truth.

Table 1 shows Twitter dataset used in our experiment.

We compared the performance of the proposed method with that of Collusionrank [15] presented in WWW 2012. Collusionrank lowers the influence scores of users who connect to spammers and filter out those users who gain high rankings by link farming. It is a user-ranking algorithm based on PageRank. Since we used the same dataset as Collusionrank, we compare the performance of the proposed method with the true positive and false positive results of Collusionrank. According to [15], Collusionrank detected 94% of the 41,352 spammers that appeared in the last low ranking scores 10% of ranking positions; consequently, we could extract the false positives of legitimate users (9.9%) from Collusionrank with a detection threshold of 10%. We reiterate that the detailed

Table 2: Performance estimation of Collusionrank [15]

	True Positive	False Positive
Spammer	94.0%	6.0%
Legitimate user	90.1%	9.9%

Table 3: Average indegree and outdegree

	Average indegree	Average outdegree
Spammer	303.6	866.5
Legitimate user	401.5	462.0

performance of Collusionrank is not described in [15], except for the true positives for spammers within the threshold of the last 10%. Table 2 is the estimated performance of CollusionRank from a true positive value of 94%.

Collusionrank has good performance in terms of true positive and false positive, but it has some limitations as follows:

First, it needs to analyze every node and edge in a social network. The PageRank-based algorithm typically estimates every node's reputation or ranking depending on the reputation of other nodes and edge formation. However, to classify spammers, computing ranks on every node is not practical. In real SNSs, spammers disseminate spamming contents simultaneously. Therefore, a real time spam filtering approach is more effective; fast spam filtering significantly decreases the number of victims of spam. As such, analyzing all social network information is not very pragmatic.

Second, it has a high proportion of false positives in detecting legitimate users. If 9.9% of legitimate user accounts in Twitter were blocked, most people would stop using Twitter. A high number of true positives in detecting spammers is also crucial; but the credibility and reliability of the service are maintained by keeping the number of false positives low.

In the following sections, we propose cascaded social information-based spam detection mechanisms that overcome the limitations of Collusionrank.

3.4. Indegree and Outdegree of Sample dataset

Since *Follow spam* has a link farming property that involves creating many outlinks, we should investigate whether spammers in Twitter have a higher outdegree than legitimate users. Also, based on Twitter's spam policy, we focus on the ratio of the indegree to the outdegree for both legitimate users and spammers.

In this paper, we use randomly selected 1,000 legitimate users and 1,000 spammers as the experimental dataset. We determined a large enough sample size with a 95% confidence level and 5% confidence interval.

Table 3 is the average indegree and outdegree of legitimate users and spammers.

Inevitably, spammers tend to have approximately two times as many out-degrees as legitimate users. The most interesting observation is that the ratio

Table 4: Performance evaluation using only indegree and outdegree

Classifier	Type	True Positive	False Positive
J48	Spammer	83.9%	16.1%
	Legitimate user	80.7%	19.3%
RandomForest	Spammer	80.8%	19.2%
	Legitimate user	80.4%	19.6%

between the average indegree and outdegree shows significant differences between legitimate users and spammers. The average indegree and outdegree of legitimate users are similar and the ratio between the two is 0.86. However, the ratio between the average indegree and outdegree of spammers is 0.35. This indicates that the indegree and outdegree could be roughly informative for classifying spammers.

To classify spammers by only indegree and outdegree, we used J48 and RandomForest classifiers built in Weka[5]. Both algorithms are decision tree based classifiers. While J48 generates only one decision tree, RandomForest corrects overfitting problems by constructing multiple decision trees during the training process. Table 4 is the classification performance evaluation using only indegree and oudegree.

As mentioned in the Twitter spam policy, we proved that the number of outdegrees can be a highly useful feature for spam classification. However, comparison using only the number of degree types between *Follow spams* and legitimate users is not enough of a performance measure to inspect spammers as shown in Table 4. To make up for the spam detection issue, we tried to apply TSP and SS as described in section 4 and 5.

4. Twitter-spam Detection with Triad Significance Profile

4.1. Follow Spam Detection with Triad Significance Profile (TSP-Filtering)

A prior study showed that, interestingly, several types of networks from different fields such as biology and the social sciences share common properties. In particular, [29] showed that some of the 13 isomorphic triad types are over-represented while some are under-represented. To the best of our knowledge, we first used this fact to discern Twitter *Follow spam*. In terms of a social graph, a user is a node and a follow from a person to another person is a directed link from the follower (the person) to the followee (another person). Figure 2 shows the 13 isomorphic triad classes introduced by [38].

Note that a *Follow spammer* inevitably generates many follows (directed links) to receive follow-backs (redirected links). For each spammer, we found all of the corresponding triads and counted the frequency of the 13 isomorphic triad classes (for detailed representation of the triad classes, refer to Figure

---

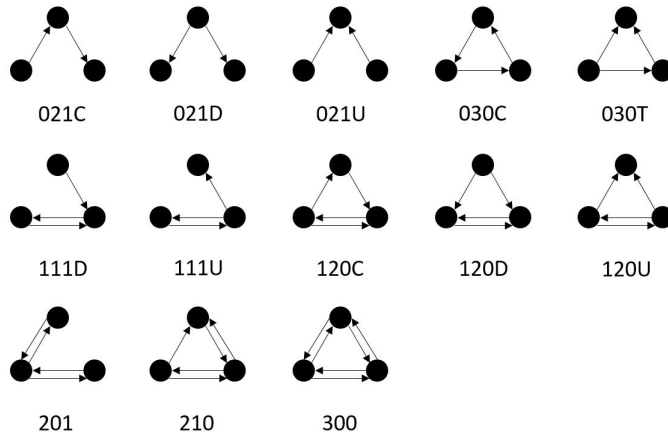[5]http://www.cs.waikato.ac.nz/ ml/weka/

Figure 2: 13 isomorphic triad classes for analyses

2). We performed the same procedures with legitimate users and compared the differences between the frequency of each triad class for both the spammer-centric triads and the legitimate user-centric triads. We argue that the triad frequencies of real social networks are different from those of spammers. The triad frequencies of spammers are similar to those of random networks with the same graph properties including the average indegree and the average outdegree.

For a given local network G_u of a user u as shown in Figure 3, we estimated the number of occurrences for each triad class. G_u consists of social links between u and 1-hop neighborhoods of u. Suppose that u is following r_1, r_2 and r_3 and is also followed by r_r, r_5 and r_6. In this case, r_1, r_2 and r_3 are "Followees" of u. In the same manner, r_4, r_5 and r_6 are "Followers" of u. Also, there are directed social links between them (represented as red-colored links in Fig. 3). To determine whether user u is a spammer or not, we analyzed user u's social graph G_u consisting of 7 nodes and 10 edges. This is a subgraph of a Twitter social network, and every user can have his/her own social network.

To discover the phenomenon whereby spammer social networks comprise subgraph features that are different from legitimate user social networks, we compared spammer triad frequencies with those of legitimate users. For each triad class i, the statistical triad occurrence is described by the *Z-score* Z_i [29] in Equation (2).

$$Z_i = (N_{spam_i} - <N_{legit_i}>)/std(N_{legit_i}) \qquad (2)$$

where N_{spam_i} is the occurrence number of the triad class i in a spammer's network, and $<N_{legit_i}>$ and $std(N_{legit_i})$ are the mean and standard deviations of its appearances in the legitimate user networks, respectively. The TSP is therefore the vector of the *Z-scores* that are normalized to length 1 in Equation
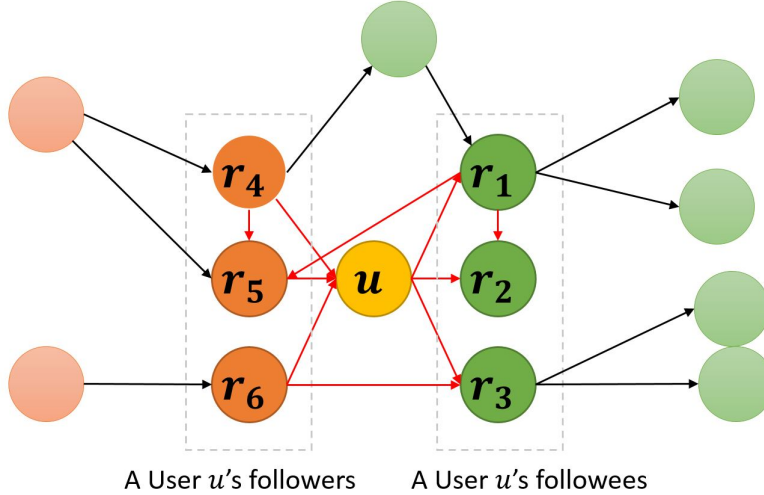
Figure 3: A user u's social network graph G_u (red-colored edges and named nodes)

(3)

$$TSP_i = Z_i/(\sum Z_i^2)^{\frac{1}{2}} \tag{3}$$

To visualize this insight from network comparison, we computed the average vector of TSP for random 1,000 spammers from the original dataset [15] and normalized it. Similarly, we also computed N_{legit_i} based on random 1,000 legitimate users. We determined that the sample size 1,000 was large enough with 95% confidence level and 5% confidence interval.

Figure 4 compares the TSPs of spammers and legitimate users. Legitimate users generally have more triads compared to spammers, meaning that the neighbors of legitimate users are socially well connected with isomorphic triad patterns; therefore, this phenomenon produced more triad counts overall. Alternatively, spammers have lower triad counts than legitimate users because their 1-hop neighbors are not likely to acquaint themselves with the other 1-hop neighbors.

Since spammers usually select their followees randomly, there are few connections between spammers' neighbors. Triad 021D, however, indicates exceptional triad counts, whereby spammers have more 021D triads than legitimate users. The 021D triad class represents the plural-following actions from a node. It also represents link-farming activity. Since the actions of *Follow spammers* involve the production of numerous out-links, their high 021D triad counts make sense. The distinction between the TSPs of spammers and legitimate users therefore explains why our TSP-detection approach is feasible. We give a statistical analysis of sensitivity of every proposed strategy in discussion section (Section 7). In the following section, we provide a true-positive rate and false-negative rate
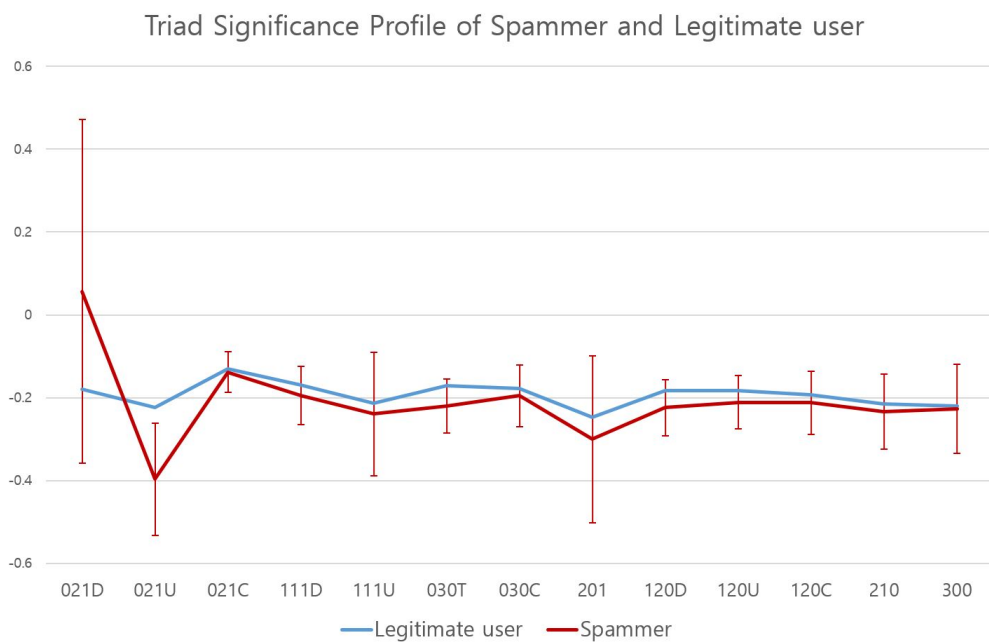
14

Figure 4: Average TSP of spammers and legitimate users. Error bar means standard deviation of spammers TSPs.

to support the excellence of this method.

As mentioned earlier, we randomly sampled sets of 1,000 spammers and 1,000 legitimate users from the original dataset [15] and conducted an experiment with TSP. We determined that the sample size was large enough with 95% confidence level and 5% confidence interval.

4.2. TSP-Filtering

The following process was used for the applicable value of the TSP-Filtering based on the experiment. First, we obtained the mean and standard deviations of each frequency for the triad class across all of the Twitter accounts. Since 1,000 legitimate users are sufficiently representative to support every Twitter account (confidence level: 95%, confidence interval: 5%), we computed the mean and standard deviations of the 1,000 randomly-sampled legitimate users. The mean value of the triad class i is $< N_{legit_i} >$ and standard deviation of the triad class i with $< N_{legit_i} >$ is $std(N_{legit_i})$, respectively. Figure shows the sampled users local social networks and triad frequency normalization.
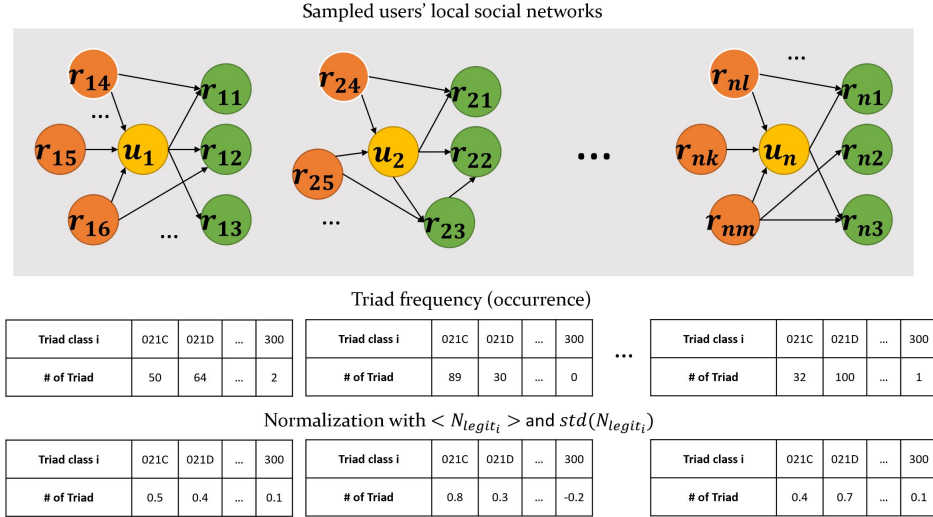

Figure 5: Triad frequency normalization

Second, we counted the spammer-triad frequencies and the legitimate user-triad frequencies for every social-network subgraph of every user account; the triad frequency represents the triad appearances in each user network. Then, we normalized the frequencies with $< N_{legit_i} >$ and $std(N_{legit_i})$ (Figure 5). In the case of the spammer, we can use Equation (2); however, in the legitimate user's case, we can use the re-translated Equation (4), where N_{spam_i} is the occurrence

Table 5: Performance evaluation using TSP-Filtering (w/o indegree and outdegree)

Classifier	Type	True Positive	False Positive
J48	Spammer	91.0%	9.0%
	Legitimate user	90.6%	9.4%
RandomForest	Spammer	92.1%	7.9%
	Legitimate user	91.6%	8.4%

Table 6: Performance evaluation using TSP-Filtering (w/ indegree and outdegree)

Classifier	Type	True Positive	False Positive
J48	Spammer	91.7%	8.3%
	Legitimate user	90.8%	9.2%
RandomForest	Spammer	92.3%	7.7%
	Legitimate user	92.4%	7.6%

number of the triad class i in a legitimate user's network:

$$Z_i = (N_{legit_i} - <N_{legit_i}>)/std(N_{legit_i}) \qquad (4)$$

Lastly, we computed each user's TSP using Equation (3). A user's TSP comprises 13 Z-scores of 13 triad classes. These 13 Z-scores could be informative for a machine-learning mechanism. We also added two features for machine learning, namely the indegrees and outdegrees of each user based on the motivation experiments.

4.3. Performance evaluation of TSP-Filtering

We conducted the experiment using J48 and RandomForest implemented in Weka (10-fold validation). Table 5 shows the performance evaluation results for the TSP method without indegrees and outdegrees. Table 6 shows the performance evaluation results for the TSP method with indegrees and outdegrees. On the other hand, Table 6 shows the performance evaluation results for the TSP method with indegrees and outdegrees.

From Table 5, even without indegrees and outdegrees, TSP-Filtering for RandomForest has a powerful spam-classification performance with 92.1%. From Table 6, the proposed approach with indegrees and outdegrees has 92.3% true positives and a lower proportion of false positives (7.6%) than Collusionrank (9.9%). Unlike Collusionrank, which needs to analyze every link to rank every node, our TSP approach is a fast and low-cost detection mechanism that uses only the 1-hop-neighborhood network for each user. Therefore, the TSP approach is a more lightweight and efficient mechanism for detecting *follow spammers* in real time.

To define a preferred sequence of attributes, we measured the importance of feature attributes based on information gain as shown in Table 7. In Table 7,

Table 7: The importance of feature attributes based on information gain (TSP-Filtering)

Feature attributes	Information Gain
021D	0.2867
021U	0.2556
021C	0.2366
111U	0.2267
201	0.1418
030T	0.1408
111D	0.1399
120D	0.136
120U	0.1075
120C	0.0871
300	0.0859
210	0.0794
030C	0.0465

feature attributes listed in descending order of information gain. Information gain can be computed as follows:

$$InformationGain(C, A) = Entropy(C) - Entropy(C|A) \qquad (5)$$

In Equation (5), C represents the given class such as spammer and legitimate user. A is the feature attribute. For example, *InformationGain(spammer,021D)* refers to the amount of entropy decrease in a spammer class when the feature attribute 021D is provided.

As we showed in the experiment results, 021D is the most significant factor in classifying *follow spammers* because of its property of two out-edges. *Follow spammers* tend to have many out-edges to legitimate users. This tendency is presented naturally in 021D. The following attribute, 021U, is also significant in classifying legitimate users because of its two in-edges. Legitimate users are likely to have more followers than spammers at stable points of the Twitter SNS system. Twitter is a very special SNS due to its subscription characteristics. The more informative the users' contents are, the more followers subscribe to the user. Since most spammers upload advertisements or spamming contents on their account, they have fewer followers than legitimate users. Understandably, some legitimate Twitter users try to follow many users at the initial and transition points for subscriptions or other reasons. However, legitimate users at the steady point have a larger number of indegrees (i.e., followers) than outdegrees (i.e., friends) due to effective influence or fruitful contents because of the psychology of popularity. In addition, the remaining features of TSP are gradually reflected in the distinction between the *follow spammers* and legitimate users because of the social-interaction.

5. Twitter-spam Detection with Status Theory

5.1. Follow Spam Detection with Social Status (SS-filtering)

Based on signs of directed links derived from social media sites such as Epinions, Slashdot, and Wikipedia, social status theory is first applied to predict certain kinds of social relationships [27]. To find strong consistency in how the model fits the data across other social networks as well as the power of influence in Twitter using our TSP filtering scheme, we additionally propose SS-Filtering for Twitter network analysis. Twitter has a special characteristic whereby users follow (or subscribe to) other users and this can be translated into the social-status theory. Generally, most Twitter users follow users who are more influential than themselves. Especially on SNS, a more-influential user is similar to a user with a high social status. When a legitimate user follows others with a higher social status, is a spammer's following pattern similar to a legitimate user? We focused on the fact that spammers are likely to follow the properties of a random network.

Our social-status-based intuition is derived from the following question: *"Is the following pattern of a spammer similar to that of a legitimate user when a legitimate user follows others with a higher status?"* We focused on the fact that spammers are likely to follow the properties of a random network. In fact, spammers have little knowledge of the social relations between legitimate users. They tend to select target users randomly. One may argue that a strong spammer is able to gather information regarding the social relations of legitimate users. However, such social engineering has only partial and instantaneous influence compared to the average and apparent influence of legitimate users.

5.2. SS-Filtering

A social-status spam-filtering system can compute the following metric based on user u's 2-hop social network. To apply Twitter to the status theory, we defined the status of a user u as the ratio of the indegree ($indegree(u)$) to the outdegree ($outdegree(u)$) of the user as shown in Equation (6).

$$status(u) = indegree(u)/outdegree(u) \qquad (6)$$

We then defined a positive link in Twitter as the probability that the user follows another user of a higher status. Figure 6 shows the concept of social status, positive link and negative link in status theory [27] and social balance theory [7, 19, 20]. A positive link, '+' link means that a node X links to each node A, B, C and D, who has higher social status than X. On the other hand, a negative link, - means that the node D links to a node X, who has lower social status than D.

The following Equation (7) is the expression of the positive link probability (PLP) of a user u:

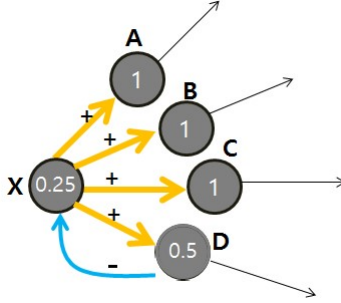$$PLP(u) = N_{pos}(u)/(N_{pos}(u) + N_{neg}(u)) \qquad (7)$$

19

Figure 6: Social status, positive link (orange link) and negative link (blue link)

Table 8: Average values comparison of social status-related features between legitimate user and spammer

Social status-related features	Legitimate user	Spammer
Average status of a user	1.82	0.39
Average positive link probability (PLP)	0.83	0.91
Average status of followees ([0-1] scale)	0.05	0.0004

where $N_{pos}(u)$ means the number of positive links and $N_{neg}(u)$ means the number of negative links. We assumed that negative links include links between same status. Additionally, we also consider the average status of followees. Table 8 compares social status-related features between a legitimate user and spammer.

We can derive interesting observations from Table 8. First, the average status of a spammer is significantly lower than that of a legitimate user, and this is attributed to the spammer's link-farming property. In this observation of the average status of a user, the two proposed schemes (TSP filtering and SS filtering) have something in common. Second, the *PLP* of a spammer to obtain a greater number of reciprocal followings is higher than that of a legitimate user. Actually, this result counters our assumption that spammers select followees randomly and the *PLP* of a spammer would be lower than that of a legitimate user. However, this result is mainly due to the significantly low status of spammer accounts (i.e., a good many outdegrees of each spammer account). We arrived at this conclusion from comparison of the average status of followees. In Table 8, we computed the average status of followees by [0-1] scale normalization for measurable comparison. Compared to the average status of legitimate users' followees, spammers' followees have definitely lower status. Therefore, though the *PLP* of spammers is higher than that of legitimate users, most spammers follow users with low status. This provides sufficient evidence for our assumption. This intuitively indicates that a spammer usually targets users who are not highly influential due to lacking real social networks. Alternatively, a legitimate user follows (or subscribes to) influential users or their

Table 9: Performance evaluation using SS-Filtering (w/o indegree and outdegree)

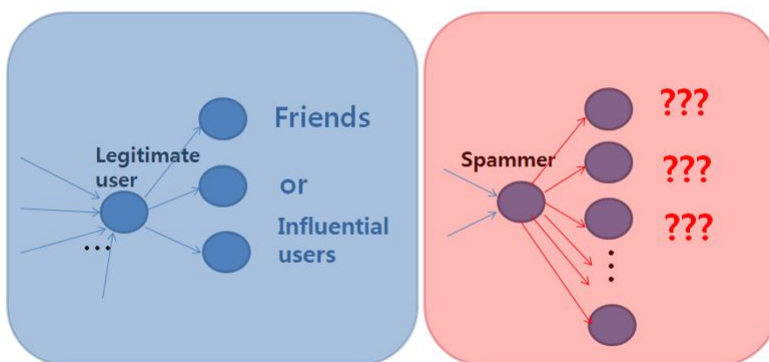| Classifier	Type	True Positive	False Positive
J48	Spammer	90.4%	9.6%
	Legitimate user	82.9%	17.1%
RandomForest	Spammer	88.6%	11.4%
	Legitimate user	85.4%	14.6%

online friends as shown in Figure 7.

Figure 7: Comparison between legitimate users and spammers for average status of followees

Figure 8 shows the relationship between an average followee status and the *PLP*, demonstrating wide variations among the average followee statuses of legitimate users. This illustrates that legitimate users follow the properties of a real social network, whereas spammers do not.

5.3. Performance evaluation of SS-Filtering

For this experiment, we used the same dataset containing samples of spammers and legitimate users from the previous section. We used the user's status, average status of followees, *PLP*, indegree, and outdegree as the feature vectors. We conducted the experiment using J48 and RandomForest in Weka (10-fold validation). The following Tables (Table 9 and Table 10) are the result of the performance evaluations of SS-Filtering considering indegree and outdegree (i.e., without or with).

Consequently, the proposed approach also shows a good proportion of true-positives (91.9%) and a lower amount of false positives (9.7%) compared to Collusionrank. Similar to the TSP method using the 1-hop-network of each user, the SS method uses only the small 2-hop-network for each user. These lightweight schemes make the spam-filtering process much faster. Considering that Collusionrank uses every link and node for computation, our detection
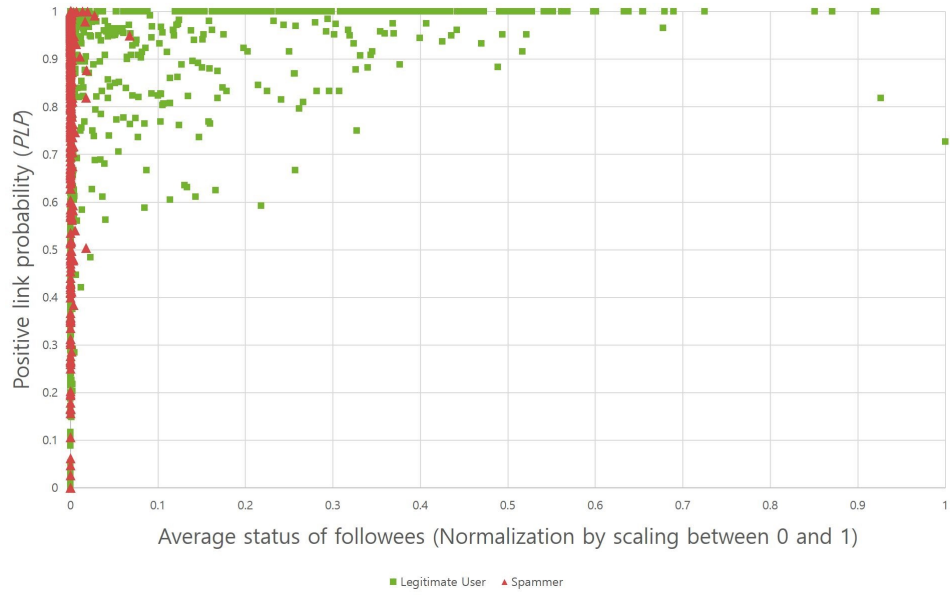
Figure 8: Green squares indicate legitimate user accounts and red triangles mean spammer accounts.

Table 10: Performance evaluation using SS-Filtering (w/ indegree and outdegree)

Classifier	Type	True Positive	False Positive
J48	Spammer	88.0%	12.0%
	Legitimate user	90.0%	10.0%
RandomForest	Spammer	91.9%	8.1%
	Legitimate user	90.3%	9.7%

Table 11: The importance of feature attributes based on information gain (SS-Filtering)

Feature attribute	Information Gain
A user's status	0.386
Followee's status	0.327
PLP	0.107

Table 12: Performance evaluation using Cascaded-Filtering

Classifier	Type	True Positive	False Positive
J48	Spammer	94.0%	6.0%
	Legitimate user	92.4%	7.6%
RandomForest	Spammer	96.3%	3.7%
	Legitimate user	94.3%	5.7%

mechanism using the social status is as efficient as the TSP approach in real time spam filtering. Table 11 shows the importance of feature attributes based on information gain with Equation (5).

In SS-Filtering, the user's status is the most significant feature similar with TSP-Filtering. This is because spammers have fewer indegrees than outdegrees. Therefore, the status of a spammer is lower than that of a legitimate user. However, the followee's average status was also important as well as the user's status. Normally, a legitimate user subscribes to informative users' accounts, and the status of these users is likely to be high due to their many followers. This means that legitimate users follow users with higher status than themselves. On the other hand, spammers tend to select and follow legitimate users randomly. As a result, their followees (targets) may have a lower status than themselves.

6. Twitter-spam Detection with Cascaded approach (Cascaded-Filtering)

In previous sections, we proposed TSP-Filtering and SS-Filtering using partial information (i.e., up to the 2-hop social network of a user) for lightweight and real-time spammer detection. Both algorithms have fewer false positives than Collusionrank, but their true positives are not superior to Collusionrank. Therefore, we suggest a hybrid approach (Cascaded-Filtering) that utilizes every feature attribute used by both TSP-Filtering and SS-Filtering. We conducted an experiment on each of 1,000 legitimate user and 1,000 spammer account with TSP features (TSP of 13 triad classes), social status features (the user's status, average status of followee, *PLP*), indegree and outdegree. Table 12 shows the performance evaluation results using Cascaded-Filtering.

7. Discussion

7.1. Overall Performance Comparison

In this paper, we compared three *Follow spam* filtering mechanisms (TSP-filtering, SS-filtering and Cascaded-Filtering) with Collusionrank. Collusion-

Table 13: Overall performance comparison

	Collusionrank	TSP-Filtering	SS-Filtering	Cascaded-Filtering
True Positive (Spammer)	94%	92.3%	91.9%	96.3%
False Positive (Legitimate user)	9.9%	7.6%	9.7%	5.7%

rank is the first *Follow spam*-targeted filtering algorithm published. It is a PageRank-based algorithm, so it can be applied when the spam-filtering system contains information on every Twitter social network. We propose the TSP-filtering and SS-filtering methods, both of which can be applied with only the 2-hop social network of a user; this is the most powerful feature of these methods. Table 13 shows the overall comparison of the three methods with Collusionrank.

Both TSP-filtering and SS-filtering have lower true-positive rates and more favorable false-positive rates. However, we are convinced that our detection mechanism is more effective for the following reasons. In general, for SNSs such as Twitter and Facebook, real-time spam detection is the most important issue. Since spammers simultaneously disseminate numerous spamming contents to SNSs, fast and immediate filtering with minimal information is needed to prevent spamming. To detect spam contents promptly, a lightweight computational cost is essential. TSP-filtering and SS-filtering identify spammers by using small subgraphs with up to 2-hop social networks for a user. On the other hand, Collusionrank requires more than 24GB of RAM to perform computation on a dataset with 1,963,263,821 edges. Therefore, for application to the entire Twitter-user population, it is not efficient to use every node and edge. We therefore suggest a divide-and-conquer approach like our TSP-filtering and SS-filtering methods for immense SNSs.

Another issue is the false-positive rate. For the reliability and convenience of SNSs, a spam-filtering system should not filter legitimate users as spammers, since this blocks users' utilization abilities and leads to notoriety and systemic failure. In comparison with the false-positive rate, compensation of the true-positive rate of TSP-filtering and SS-filtering is natural and easier. Spam-reporting services are incorporated into the design of most SNSs for the detection of content abusers, and this complementary tool could be helpful for the detection of subtle spamming actions. Accordingly, TSP-filtering and SS-filtering could be practical spam-filtering mechanisms for use under SNS conditions.

The performance of Cascaded-Filtering, which employs every feature attribute used in TSP-Filtering and SS-Filtering, is superior to the other three schemes including CollusionRank. This scheme can accurately detect more spammers and block or suspend less legitimate users. Consequently, this result supports the idea that there is a distinction between the legitimate user's

Table 14: True Positive, False Negative and Recall of every suggested strategy

	True Positive	False Negative	Recall
In/Out degree	0.808	0.804	0.501
TSP-Filtering	0.923	0.924	0.500
SS-Filtering	0.919	0.903	0.504
Cascaded-Filtering	0.963	0.943	0.505

social network and the spammer's social network. Especially, this scheme does not require the full social network information, including users that are not directly related to the account, to determine whether a specific user is a spammer or not. Moreover, for the service provider, half the number of false positives of Collusionrank is pretty attractive to ensure a reliable and convenient SNS system.

(a) TSP-Filtering
AUC = 0.9727

(b) SS-Filtering
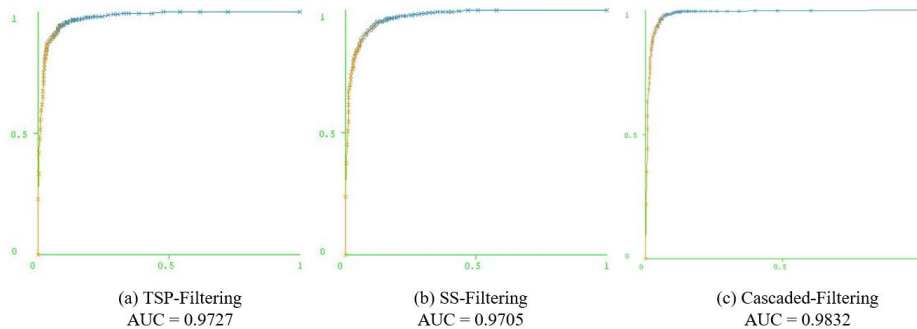AUC = 0.9705

(c) Cascaded-Filtering
AUC = 0.9832

Figure 9: ROC curve of (a) TSP-Filtering and (b) SS-Filtering (c) Cascaded-Filtering (X axis : False positive , Y axis : True positive)

Figure 9 shows the Receiver Operating Characteristic (ROC) curve and the Area Under the ROC Curve (AUC) for every proposed approach. Due to its high true positive and low false positive values, Cascaded-Filtering has the highest AUC. To compare with Collusionrank, we estimated the AUC of Collusionrank using its true positive and false positive values. Consequently, compared to Collusionrank (AUC=0.92), our approaches are competitive and require much less social network information.

7.2. Sensitivity Analysis

Figure 10 shows a statistical analysis to evaluate the sensitivity of the TSP-detection strategy.

We computed the recall(sensitivity) as following equation :

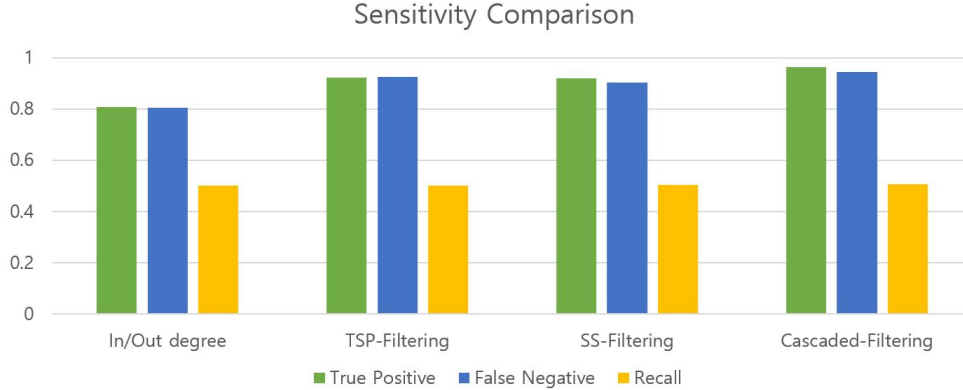$$recall = TruePositives/(TruePositives + FalseNegatives) \qquad (8)$$

25

Figure 10: Sensitivity of the proposed detection strategies

To sum up, from table 14, two proposed approaches, SS-Filtering and Cascaded-Filtering, show better sensitivity than spam classification using In/Out degree only. Actually, TSP-Filtering has slightly lower sensitivity than those approaches, it counters this limitation with both high true positive and false negative. Cascaded-Filtering, which is the hybrid approach using every feasible features, shows the best performance at sensitivity with superior true positive and false positive.

7.3. Complexity Analysis

Following pseudo-code is TSP-Filtering and it consists of two steps: Triad census algorithm and Triad Significance Profile computation algorithm. At first, we modified input of Triad census algorithm [2] to apply to two-hop input subnetwork. $G = (V, E)$ is the two-hop directed subnetwork of a user.

Algorithm 1 computes frequencies of 16 isomorphic triad types. In our work, we used only 13 isomorphic triads because triad types with isolated nodes are counted frequently so that this interrupts observing significance of triads. From this algorithm, we can get frequencies of 13 isomorphic triad types of given graph G. TSP-Filtering get 2-hop neighborhood social network of a certain user as input. This algorithm has the complexity $O(m)$ and m is the number of edges in the graph G. In algorithm 1, TriType means how many nodes are connected each other. This indicates that an isolate node exists when TriType is 2. [2] used the concept of Tricode to count triad efficiently, but we dont explain detail of the concept in this paper.

Algorithm 2 computes Z-score and TSP of a certain input user. For input attributes, this algorithm has input vectors consisting of mean and standard deviation of legitimate users triad frequency. These attributes could be got at preprocessing phase, which computes mean and standard deviation of legiti-

---
Algorithm 1 TriadCensus

$INPUT : G$
for i:=1 to 16 **do**
    $N_i = 0$
end for
for $v \in V$ **do**
    if $u \in v$ **then**
        $S:= Neighbor(u) \cup Neighbor(v) \setminus \{u,v\}$
        if Link(u,v) and Link(v,u) **then**
            TriType:=3
        else
            TriType:=2
        end if
        N[TriType]:=N[TriType] + n - |S| - 2
        for each $w \in S$ **do**
            if $u < w \vee (v < w \wedge w < u \wedge \neg Link$(v,w) **then**
                TriType:=TriType[Tricode(v,u,w)]
                N[TriType] := N[TriType]+1
            end if
        end for
    end if
end for
sum:=0
for i:=2 to 16 **do**
    sum:=sum+$N[i]$
end for
$N[1]$:= $(1/6)n(n$-1$)$-sum
return N

Algorithm 2 Triad Significance Profile

$INPUT : N, < N_{legit_i} >, \text{std}(N_{legit})$
for i:1 to 13 **do**
    Z_i=$(N[i]$-$< N_{legit_i} >)$/std(N_{legit_i})
end for
for i:1 to 13 **do**
    TSP_i=$Z_i/(\sum Z_i{}^2)^{\frac{1}{2}})$
end for
return TSP

27

Table 15: Complexity analysis

	Collusionrank	TSP-Filtering	SS-Filtering	Cascaded-Filtering
Time complexity	O(N+M)	O(m)	O(n)	O(m+n)
Space complexity	O(N)	O(m)	O(n)	O(m+n)

mate users for the training value. Since our work is to classify spammers from legitimate users, we should compare the users triad frequency N (testing value) to legitimate users triad frequency N_{legit} (training value). This algorithm has time complexity O(1) because of a simple normalization process. Therefore, TSP-Filtering is a combined algorithm of TriadCensus algorithm and TSP algorithm. The novelty of our work is that we used a users 2-hop neighborhood social network as initial input. Also, we compare each users census to overall legitimate users to observe overrepresentation and underrepresentation in each isomorphic triad. Overrepresentation and underrepresentation of triads could be the distinct features to classify spammers from legitimate users. So, TSP-Filtering has O(m) time complexity for a target user. Following pseudo-code is SS-Filtering. Actually, in preprocessing phase, we can compute the status of a user with his/her indegree and outdegree. Since we defined the status of a user as the ratio of the indegree to the outdegree of the user (Equation (6)), we just update three values indegree, outdegree and status in real time case. So, we only need positive link probability (PLP) of a user. The simple algorithm for PLP is as follows. $Status$ means a table with status of every user in G.

Algorithm 3 Positive Link Probability

$INPUT : G, Status$
$PositiveLink$:=0
for each $u \in Neighbor(v)$ **do**
   if $Status[v] < Status[u]$ **then**
      $PositiveLink$:=$PositiveLink+1$
   end if
end forPLP:=$PositiveLink/|Neighbor(v)|$
return PLP

Algorithm 3 computes the ratio of the number of neighbors with higher status than user v to the number of neighbors of user v. Since SS-Filtering highly depends on preprocessing phase, this simple algorithm has the complexity O(n) when n means the number of users in G.

Additionally, Table **??** shows the complexity analysis of the proposed schemes and Collusionrank. Actually, since our work is for each users 2-hop neighborhood social network, complexity comparison with Collusionrank seems to be ironic. To sum up, Collusionrank is for classifying large number of spammer accounts from whole social network, but TSP-Filtering, SS-Filtering and Cascaded-Filtering is for determining if a user is spammer or not. n means the number of user (nodes) in 2-hop neighborhood social network and m is the number of relation (edges) in 2-hop neighborhood social network. Also, N means

the number of every users (nodes) in whole social network and M means the number of every relations (edges) in whole social network. This table showed that our approaches need less computation than Collusionrank.

8. Future Work

8.1. Dynamic follow spammer detection

First of all, camouflage attack, which uses compromised user account to disseminate spam contents, could be one of the interesting future work. To solve the camouflage attack problem, unsupervised scheme can be considered for detecting newly occurred spammers to the proposed scheme (i.e., unsupervised scheme based on cascaded social information). For example, we can select self-organizing map (SOM) among machine learning algorithms for mapping between spammer labeled with supervised scheme based on cascaded social information and new spammer based on unsupervised scheme.

In detail, SOM is one of clustering algorithms using Euclidean distance concept between source and destination. If we train spammer patterns with cascaded social information scheme as a supervised concept that can be a partial part of pre-defined map of SOM for processing and clustering new input automatically as a unsupervised concept. SOM calculates Euclidean distance between spammer account with cascaded social information and new Twitter account to check whether the tendency of new account is closer to spammer or legitimate. In each epoch, SOM can update the tendency of each account based on Euclidean distance dynamically and distinguish between time-varying spammer and legitimate based on clustering map. In future, we will apply SOM or other optimal solution for detecting dynamic spammer according to time.

8.2. Generalized applications of the proposed scheme

In the view of social behavior, people have different social networking style based on their purpose of using SNS. For example, spammers make numerous following links to unspecified individuals with the aim of spreading spam contents. On the other hand, if someone wants to make friends in SNS, they would carefully select users as friends based on a personal preference and closeness. And finally, these social links made by various users could be interpreted in cascaded social information.

As shown in our experiment, cascaded social information and its structural analysis suggest various future application in social network security. In future work, we are to find a correlation between various behavior patterns of using SNS and cascaded social information. The behavioral patterns of using SNS can be more specified by various types regardless of spammer and legitimate. One of the types is a false rumor in SNS. Since false rumor is unverified and unconfirmed information, it seems very attractive sometimes according to its degree of sensationalism. In general, false rumors are spread widely in the community and proven to be false eventually. However, we need to focus on users who spread false rumors in the view of their behavioral patterns. If someone

has high social status in SNS, he/she tends to hesitate to spread rumors because he/she is anxious for impairing his dignity or credibility by spreading misinformation. Therefore, we could guess that false rumors are spread by users who arent related to social credibility or reputation. In conclusion, false rumor detection based on cascaded social information could be one of great topics for future generalized application.

9. Conclusion

On Twitter, one of the most popular social networking services (SNSs), a new kind of spamming strategy has emerged known as Follow spam. The goal of this paper is to classify follow spammers by utilizing social network properties in the individuals local social network. To solve this problem, we proposed three novel cascaded social information based spam detection mechanisms (TSP-Filtering and SS-Filtering) and a hybrid approach (Cascaded-Filtering). These approaches analyze and exploit social network properties such as Triad Significance Profile (TSP) and Social Status (SS). We conducted large-scale experiments on real Twitter datasets. The results from analyzing individual-related small local social networks support our assumption that a spammers social network is different from a legitimate users social network. We compared our approaches to Collusionrank, the PageRank-based representative algorithm of the follow spam detection. Cascaded-Filtering was found to be the most competitive and superior approach while requiring much less social network information. In conclusion, with a high proportion of true positives (96.3%) and low amount of false positives (5.7%), our approaches are very secure and practical mechanisms that can be applied as real time spam detection systems.

Acknowledgments

References

References

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

[2] V. Batagelj, A. Mrvar, A subquadratic triad census algorithm for large sparse networks with small maximum degree, Social networks 23 (3) (2001) 237–243.

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

[4] A. A. Benczúr, K. Csalogány, T. Sarlós, Link-based similarity search to fight web spam, in: In AIRWEB, Citeseer, 2006.

[5] F. Benevenuto, G. Magno, T. Rodrigues, V. Almeida, Detecting spammers on twitter, in: Collaboration, electronic messaging, anti-abuse and spam conference (CEAS), vol. 6, 2010, p. 12.

[6] Q. Cao, M. Sirivianos, X. Yang, 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, USENIX Association, 2012, pp. 15–15.

[7] D. Cartwright, F. Harary, Structural balance: a generalization of heider's theory., Psychological review 63 (5) (1956) 277.

[8] C. Castillo, D. Donato, A. Gionis, V. Murdock, F. Silvestri, Know your neighbors: Web spam detection using the web topology, in: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2007, pp. 423–430.

[9] M. Cha, H. Haddadi, F. Benevenuto, P. K. Gummadi, Measuring user influence in twitter: The million follower fallacy., ICWSM 10 (10-17) (2010) 30.

[10] G. Danezis, P. Mittal, Sybilinfer: Detecting sybil nodes using social networks., in: NDSS, San Diego, CA, 2009.

[11] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, M. Theimer, Reclaiming space from duplicate files in a serverless distributed file system, in: Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on, IEEE, 2002, pp. 617–624.

[12] M. Egele, G. Stringhini, C. Kruegel, G. Vigna, Compa: Detecting compromised accounts on social networks., in: NDSS, 2013.

[13] S. Fakhraei, J. Foulds, M. Shashanka, L. Getoor, Collective spammer detection in evolving multi-relational social networks, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 1769–1778.

[14] H. Gao, Y. Yang, K. Bu, Y. Chen, D. Downey, K. Lee, A. Choudhary, Spam ain't as diverse as it seems: throttling osn spam with templates underneath, in: Proceedings of the 30th Annual Computer Security Applications Conference, ACM, 2014, pp. 76–85.

[15] S. Ghosh, B. Viswanath, F. Kooti, N. K. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, K. P. Gummadi, Understanding and combating link farming in the twitter social network, in: Proceedings of the 21st international conference on World Wide Web, ACM, 2012, pp. 61–70.

[16] N. Z. Gong, M. Frank, P. Mittal, Sybilbelief: A semi-supervised learning approach for structure-based sybil detection, Information Forensics and Security, IEEE Transactions on 9 (6) (2014) 976–987.

[17] P. Graham, A plan for spam (2002).

[18] Z. Gyöngyi, H. Garcia-Molina, J. Pedersen, Combating web spam with trustrank, in: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, VLDB Endowment, 2004, pp. 576–587.

[19] F. Heider, Social perception and phenomenal causality., Psychological review 51 (6) (1944) 358.

[20] F. Heider, Attitudes and cognitive organization, The Journal of psychology 21 (1) (1946) 107–112.

[21] J. Hovold, Naive bayes spam filtering using word-position-based attributes., in: CEAS, 2005, pp. 41–48.

[22] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, S. Yang, Catchsync: catching synchronized behavior in large directed graphs, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2014, pp. 941–950.

[23] I. Kayes, N. Kourtellis, D. Quercia, A. Iamnitchi, F. Bonchi, The social world of content abusers in community question answering, in: Proceedings of the 24th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2015, pp. 570–580.

[24] J. M. Kleinberg, Authoritative sources in a hyperlinked environment, Journal of the ACM (JACM) 46 (5) (1999) 604–632.

[25] V. Krishnan, R. Raj, Web spam detection with anti-trust rank., in: AIRWeb, vol. 6, 2006, pp. 37–40.

[26] H. Kwak, C. Lee, H. Park, S. Moon, What is twitter, a social network or a news media?, in: Proceedings of the 19th international conference on World wide web, ACM, 2010, pp. 591–600.

[27] J. Leskovec, D. Huttenlocher, J. Kleinberg, Signed networks in social media, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2010, pp. 1361–1370.

[28] J. Martinez-Romo, L. Araujo, Detecting malicious tweets in trending topics using a statistical analysis of language, Expert Systems with Applications 40 (8) (2013) 2992–3000.

[29] R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, U. Alon, Superfamilies of evolved and designed networks, Science 303 (5663) (2004) 1538–1542.

[30] D. O'Callaghan, M. Harrigan, J. Carthy, P. Cunningham, Identifying discriminating network motifs in youtube spam, arXiv preprint arXiv:1202.5216.

[31] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: bringing order to the web.

[32] D. Sculley, G. M. Wachman, Relaxed online svms for spam filtering, in: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2007, pp. 415–422.

[33] L. Shi, S. Yu, W. Lou, Y. T. Hou, Sybilshield: An agent-aided social network-based sybil defense among multiple communities, in: INFOCOM, 2013 Proceedings IEEE, IEEE, 2013, pp. 1034–1042.

[34] G. Stringhini, G. Wang, M. Egele, C. Kruegel, G. Vigna, H. Zheng, 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.

[35] N. Tran, J. Li, L. Subramanian, S. S. Chow, Optimal sybil-resilient node admission control, in: INFOCOM, 2011 Proceedings IEEE, IEEE, 2011, pp. 3218–3226.

[36] B. Viswanath, M. A. Bashir, M. Crovella, S. Guha, K. P. Gummadi, B. Krishnamurthy, A. Mislove, Towards detecting anomalous user behavior in online social networks, in: Proceedings of the 23rd USENIX Security Symposium (USENIX Security), 2014.

[37] A. H. Wang, Don't follow me: Spam detection in twitter, in: Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on, IEEE, 2010, pp. 1–10.

[38] S. Wasserman, K. Faust, Social network analysis: Methods and applications, vol. 8, Cambridge university press, 1994.

[39] W. Wei, F. Xu, C. C. Tan, Q. Li, Sybildefender: Defend against sybil attacks in large social networks, in: INFOCOM, 2012 Proceedings IEEE, IEEE, 2012, pp. 1951–1959.

[40] B. Wu, B. D. Davison, Identifying link farm spam pages, in: Special interest tracks and posters of the 14th international conference on World Wide Web, ACM, 2005, pp. 820–829.

[41] B. Wu, V. Goel, B. D. Davison, Topical trustrank: Using topicality to combat web spam, in: Proceedings of the 15th international conference on World Wide Web, ACM, 2006, pp. 63–72.

[42] S. Yardi, D. Romero, G. Schoenebeck, et al., Detecting spam in a twitter network, First Monday 15 (1).

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

[44] H. Yu, M. Kaminsky, P. B. Gibbons, A. Flaxman, Sybilguard: defending against sybil attacks via social networks, ACM SIGCOMM Computer Communication Review 36 (4) (2006) 267–278.

[45] B. Zhou, Y. Yao, J. Luo, Cost-sensitive three-way email spam filtering, Journal of Intelligent Information Systems 42 (1) (2014) 19–45.