Spam Account Detection - Facebook

By Mehak Mehta & Amit Khandelwal

Abstract

Advent of online social media like Facebook, Google+, Twitter, etc. has changed the way we do social interaction with the world. At the same time it has given new ways of spamming or advertising, creating millions of web pages everyday as spam and then circulating them through viral tagging and posts. Our project aims to study the behavioral patterns of our friends or connections on social network and then try to classify them based upon their interaction behavior as spammers or fake accounts.

I. Introduction

Online social networks have become a part of our life. In a sense it is most convenient way for us to interact with the world i.e. friends, family, professional contacts etc. Everybody is there and reachable. It's a virtual social world with all its pros and cons. On one hand we have friends, communities, groups etc. with whom we interact and like sharing our thoughts using messages, posts, chats, tagging etc. At the same time there are people whom we don't like or which nag us with pings, messages and unsolicited posts and tagging. At the same time we have people who does not interact with us or rarely are even active on their accounts. These accounts are like e-waste since the people who created them have forgotten about them and they are there with no purpose at all.

As online social networks like Facebook are growing the problems like spamming and fake accounts is increasing at tremendous speed. They are creating a lot of e-waste and are in a way chasing away the real users. In this project we have tried to weed out such people from our friend list who behave in such abnormal ways using the Social Interaction analysis of our accounts. This project uses the history of social interaction and behavior of users to categories whether the people in our friends list are fake/spammers or not. In order to do this, we have categorized each of our friend's relationship with us and tried to study their online social behavior.

Fake accounts on social networks like Facebook are a big problem nowadays. Apart from creating nuisance to other users, these accounts pose a serious threat to social network privacy and integrity. We have tried to target fake accounts based on inactivity and spamming behavior on Facebook. In addition, we evaluated certain side effects of these fake accounts such as fake ties, fake messages or reviews (in some cases advertisements or propaganda). We can detect these fake accounts using their behavioral patterns. The patterns are related to their profiles, activities, posts, etc. We have used two different approaches to determine which are the top nagging or inactive accounts in our network then verified the results with the account holders. We have also plotted these results on their respective ego networks to give more insight about the results we found.

In the first approach we have come up with an algorithm based on ten characteristics of a fake account on the Facebook [1]. We have used our own scoring method to assign a score to each of our friends based upon their profile characteristics and their interaction behavior with us. We have come up with the top ten accounts with highest fake account scores as results and checked them with the ground truth. Then we plotted them on their respective ego network.

In the second approach we scanned old posts, messages, tags, likes, comments etc. to come up with an interaction matrix which will depict our interaction behavior with our friends. Then we used Markov Clustering algorithm to classify the accounts as spammers or fake. Similar to the first approach we then plotted the spammers on the respective ego network.

In the end, we compared the ego graphs obtained from the both approaches to analyze the results.

II. Background & Related Work

From the early stages of online social networks, spamming has been identified as one of the major problems, which is aggravating day by day. As it is a convenient way to attract people to their company's products or to lure people into giving their credentials, or just for fun sending viral controversial posts or messages. The social networks like Facebook have become the fastest way to spread news online without spending a penny in matter of few minutes or days. It is evident that some malicious people want to take undue advantage of such phenomenon and spread their spam post for monetary reasons or otherwise.

A lot of work has been done in spam detection field, but in recent years its importance has increased a lot since people have realized the potential of social networks and are trying to misuse it. Facebook along with other educational research institutes are doing a lot of work in this area to weed out spammers and fake accounts so they can clean social networks for real users.

Various methods have been used in past for detecting spammers like machine learning algorithms, statistics, graph theories etc. For e.g. a lot of research papers in this area has used unsupervised learning algorithms [3,4]. In both of these papers, authors have used the spam data from malicious accounts to do unsupervised learning. By this, they can detect or classify future posts by any account as spam or not. In [4], UNIK system goes one step further by leveraging information from user profiles and their behavior to detect spammers. Our project approach is also inspired by the UNIK system to detect spammers. But it also tries to categories them as fake or not. Moreover, we used another methodology based upon Markov Chain to detect spammers, which in our understanding can be more effective when used for profile classification.

Another approach employed by many researchers is to make Honey-profiles [5] to attract spammers to connect and then study their social behavior and profiles. This gives an insight into the minds of these spammers, which can be a very good counter technique for detecting them. Then they used a classifier to train and then classify spams as done in previous papers. In this way, more-or-less most of the work done till now is based upon employing machine learning algorithms on various available or collected datasets. Then use them to classify spam posts. But not much work has been done to study the interaction behavior between friends.

Our paper tries to study the social and interaction behavior of friends on different volunteer's networks. Then it tries to detect abnormalities in the interaction patterns like unusually high or low activity. Based on this, our project tries to understand the characteristics of interactions in social groups. The final result classifies accounts as spammers/fake using both algorithms and compares the results with the ground truth data.

III. Data Collection

We have collected data from Facebook Graph API of several accounts of our volunteering friends. Following data was collected: -

a) Profile Data (About) of friends
b) Inbox (messenger chats)
c) Wall Feed
d) Mutual Connection Network (for Ego Network)
e) Reverse Image Analysis of profile picture

We collected data from 10 people. We used this data to parse and collect the interaction data between our friends and us. This data formed the basis for calculations in the implementation of our project.

Since there are very less or rare some fake/spam accounts added in our personal Facebook profiles, we created two Honey-Profile accounts to attract spammers and fake accounts in order to collect more data from spammers. It took efforts for us to breed those accounts and presently both accounts have around 5-7 spammers added. It will take more time for us to get more data from these profiles.

IV. Implementation

A. Approach 1: Social & Interaction Behaviors

We applied two algorithms to find out the fake and spam accounts (which in our case were mainly inactive and advertiser's account) on facebook. First algorithm targets the profile data and their accounts activities and compute a final score based on the following conditions, if met: -

i. Girls generally don't put their contact number in public/social sites. However, fake profiles of girls usually have a contact number mentioned in their contact information.
ii. If most of the friends are of the opposite gender, it can be assumed that the account is used/created either for fun or for random dating. Thus, it can be strong factor to detect fake accounts.
iii. Birthday can also be the criteria. Birth dates like 1-January, 1-April, 31-December, etc. are common between fake accounts as it is quite unique and easy to type in.
iv. If most of the essential information like school, education or workplace, email ids are missing and that the user is looking for dating and interested in both men and women, it shows signs of fakeness.
v. Check the posts, comments, and likes by a particular account. If the account was highly active during a certain period and become inactive after that, there is a high probability that account is fake and created for a specific purpose, which must be fulfilled now.
vi. Check the location information like hometown, current location, email ids, etc. If this information is missing, it can act as a supporting factor for fakeness.
vii. Profile photo can be an important factor. If the profile photo is showing some signs of vulgarity or nudity, then there is a high chance of account being fake.

No single factor if true, can claim that the account is 100 percent fake or spam. Every factor either increases or decreases the chances of account being fake or real. So, we came out with an algorithm to first provide *factor value* (a numerical value) for each of the mentioned points and then for every account, we sum up all the fake factor values and came with a final *profile score.* For example, the 1st point can be a very strong factor, so it has been given the value 8, 3rd point can be a supporting factor (not very strong), so given the value of 3. Similarly, every point is given a weightage and a final score is calculated at the end by summing all the values called as **profile score.**

Figure 1 (a) shows the code snippet of how a set data structure is created for each friend in an individual's account and how profile score is calculated according to weightage of particular factor.

Figure 1 (b) shows the resulting set for each friend with number showing the presence of any above-mentioned factors

```
fake_score = [set() for i in range(0,No_friends)]
if data[i]["sex"] == "female" and data[i]["email"] != None:
        fake_score[i].add(7)
        fake_value += 8
if data[i]["current_location"] == None and data[i]["hometown_location"] == None
and data[i]["email"] == None:
        fake_score[i].add(5)
        fake_value += 3
if data[i]["profile_update_time"] < 1400000000:   ## Tue May 13 12:53:20 EDT
2014
        fake_score[i].add(9)
        fake_value +=9
```

Figure 1(a) Profile Score Calculation

```
(['Garima Gaur', 11, 5, 6])
(['Jitendra Sharma', 11, 5])
(['Prateek Chaudhary'])
(['Suhani Bansal',11,2,5,8])
```

Figure 1(b) Characteristic Factor Set

To make the final outcome (profile score) stronger, we included the interaction data as well and add or subtract their weightage to the score. Using the inbox data (retrieved from Graph API explorer), we calculated the interaction matrix of every account with all his friends. Interaction matrix includes *two-sided* communication, *one-sided* communication, *fromMe* and *ToMe* values. These values are calculated for each friend in a facebook account. The following heuristics were applied to contribute to the profile score value: -

a) A negative factor value is given if there are several two-sided communications for a particular friend.

b) If there are one-sided communications, then total number of FromMe (particular account's person messaged a friend in his friend-list but without a reply) and ToMe(message from a friend but without a reply from that particular account) communications are calculated. FromMe communication decreases the weightage on fakeness whereas ToMe count bolsters the fakeness factor.

Fake_Score = Fake_Score - 2*Two-Sided + 3*One-Sided – 0.1*FromMe + 0.3*ToMe (1)

Equation (1) is the formula applied to add the interaction information to the profile score.

Another valuable information is the wall interaction that we used to add to the score. We retrieved the **feeds** data for the same from Graph API explorer. The following criteria were applied to contribute to the fake score: -

1. Likes of a particular friend on the wall.
2. Number of Tags of that friend.
3. Total number of comments by him.
4. Number of Message-Tags done by him for you and vice versa.

Fake_Score = Fake_Score + 2*Likes + 3*Tags – 3*Comments - 0.1*MessageTag (2)

The figure 2 (a) (b) shows the resultant interaction matrix (combined with wall interaction) for each friend of an account. The fake score is updated out of these 8 values as per the equations (1) and (2).

```
Muskan Mehta  [1 1 0 1] [2,1,2,0]
Hema Khatri    [1 3 0 22] [0,0,0,0]
Suprit Todwal  [1 1 0 10] [1,0,2,0]
Amogh Gupta   [40 1 0 1] [12,10,32,2]
```

Figure 2(a) Interaction Matrix

```
Maximum score people:-
Ashish Singh 39.0
Apurv Jain 37.0
Nikhalesh Khandelwal 29.0
Anoop Kumar 26.0
Sinjan Kumar 20.0
Anirudh Ghanta 18.0
Dheer Veer Vikram Singh 18.0
Vikash Kumar 15.0
```

Figure 2(b) Top Profile Scorers

Top 10 accounts with the highest profile score is the final output of this algorithm, which is then used with Networkx python library to draw and show the bi-populated Ego-Network.

There is a general trend being followed by fake or spam accounts: 'like' all the posts of the user, 'tag' a user in each of their posts. So these factors are given positive values. Real friends generally do commenting and tagging in the messages/chat, hence these values are given negative factor values.

After calculating the fake score, 10 friends with highest profile score (highest candidates for fakeness or spammers) were taken out and compared with the ground truth. For ground truth, these results were shown to the account holders to check manually and verify the results. To depict the results, we used bi-populated ego networks. The nodes with highest profile score were colored in a different color (red) than the rest of the nodes (blue). This seemed to be the best way to analyze behavioral characteristics of fake accounts pictorially. It shows their interaction with the other nodes in network, the social community they fall in and other behavioral details.

B. Approach 2: Markov Chain Clustering

In the second approach, we have modeled the social network data using a weighted graph in which user profiles are represented as nodes and their interaction as edges. The weight on each edge shows how strong the interaction between them is. Is it two sided or just one sided? In other words the interaction graph is *G (V, E, W)* where V is set of friends in a network, $E \subseteq V \times V$ is the set of edges and W is the set of weights assigned to each of them based upon amount of interaction. For each edge E between the account owner and his friend, we will find the *active* friends, *page likes* and *shared URL-tags* forming a vector which will be converted into a weight. Following are three types of social interactions we have considered: -

i. **Active Friends**:
This feature measures the interaction frequency of a user with his/her friends in the network. For a user v_i with his with set of friends as F_i, Active friends F_i^a are the subset of F_i who interacts with the user via wall post, comments or tags. Similarly, we find the set of common active friends between two friends v_i and v_j using intersection of active friends between them. For an edge $e_{ij} = (v_i, v_j)$ the value of active friends is given as:

$$F_{ij} = F_i \cap F_j$$

ii. **Page Likes:**
This feature measures the page-likes frequency of a user with his/her friends in the network. It is a subset of interaction frequency measured in active friends but gives more fine tuned data over it. So for an edge $e_{ij} = (v_i, v_j)$ the value of page-likes P_{ij} is calculated as intersection of the sets of page-likes of v_i and v_j given as:

$$P_{ij} = P_i \cap P_j$$

iii. **Shared URLs-Tags:**
This feature measures the shared-urls & tags frequency of a user with his/her friends in the network. It is also a subset of interaction frequency measured in active friends but gives more fine tunes data over it. So for an edge $e_{ij} = (v_i, v_j)$ the value of page-likes P_{ij} is calculated as intersection of the sets of shared-urls & tags of v_i and v_j given as:

$$U_{ij} = U_i \cap U_j$$

On the basis of the above three characteristics we calculate the weight on each edge $e_{ij} = (v_i, v_j)$ of our social interaction graph as w_{ij} given as:

$$w_{ij} = F_{ij} + P_{ij} + U_{ij}$$

We will use the graph *G (V,E,W)* of user's social network with nodes $V=\{v_i: i= 1, 2..n\}$ with $E=\{(v_i,v_j): 1 \leq i,\ j \leq n\}$ to calculate a adjacency matrix A_{nxn}. The value of each cell $A(i,j) = a_{ij} \geq 0$ represents the weight w_{ij}.

So this matrix represents Activity Matrix of a user and his friends.

On the matrix A, we will apply Markov clustering which uses random walk and $M(i,j)=m_{ij}$ is the transition probability. Considering the transition probability from one node to another in t steps as $M.M^{t-1}$, the transition probability is inflated. Due to this higher transition probabilities are increased and lower are decreased. This is done by taking m_{ij} to the power $r \geq 1$ as given by following equation:

$$\Upsilon(M,r) = \left\{ \frac{(m_{ij}^{r})}{\sum_{a=1}^{n}(m_{ia}^{r})} \right\}_{i,j=1}^{n}$$

The markov clustering method is performed iteratively with successive powers of M and then performs the inflation process. The iteration terminates when the matrix achieves a stable solution as:

$$|M_t - M_{t-1}| < €$$

In our experiments we performed the Markov clustering using expand factor as 2, inflate factor (r) as 1.5, max loop 100 and multiplying factor as 2. The markov clustering gave us different number of clusters for each user (around 10-15) per user. Most of the nodes were found in first cluster correspond to row 1 of out pus matrix. These are nodes having normal interaction with the user. The other nodes were dispersed in all the rows of the matrix in groups of 1-8 each. These clusters contains all the outlier nodes (or friends) having unusual social interaction either very high or very low depending upon the user being analyzed. If we have some spammers in our account the clustering will catch them as high interaction activity nodes in the inflation step of random walk.

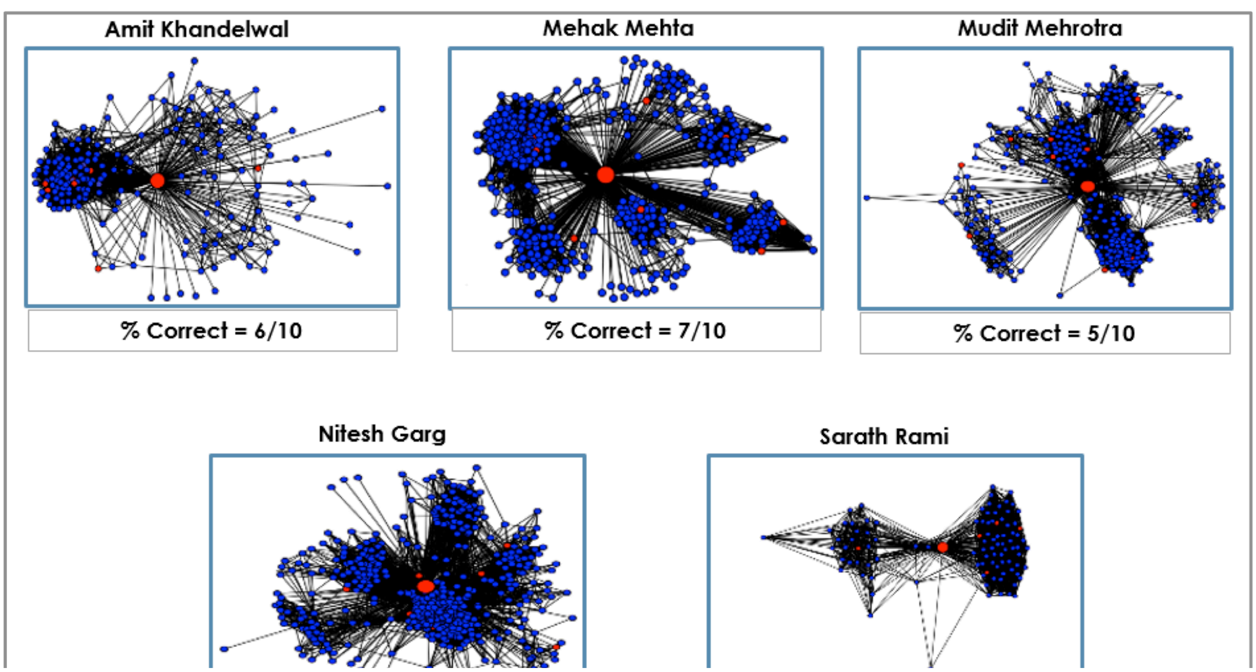
V. Results:

Result 1: Social & Interaction Behaviors

The results are shown pictorially on a bi-populated Ego Network with *red nodes* showing the highest characteristics of spammers (or fake accounts) which are having unusual high or low activity and all the others nodes are depicted as *blue*.

We confirmed our results with the ground truth by requesting the account holders to confirm us by checking the behavior of these 10 nodes manually. The percentage correctness of each account is shown in the Figure 3. The overall accuracy we obtained from the approach 1 was **56%.**

Figure 3. Ego Networks from Approach: Social & Interaction Behaviors



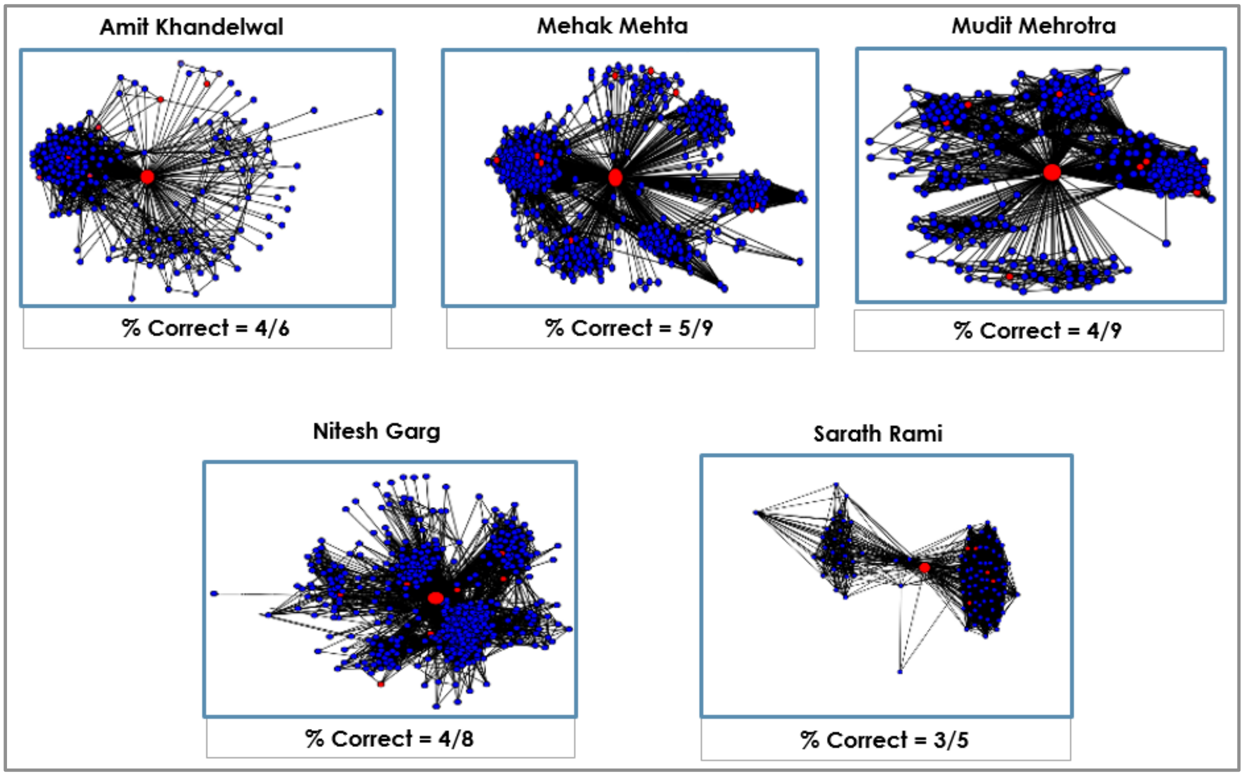


Figure 4. Ego Networks from Approach: Markov Chain Clustering

Result 2: Markov Chain Clustering

The Markov clustering results gave around 10-15 clusters as outliers. 1-2 clusters contain the majority of nodes with normal interaction social activity. All the other clusters contained outliers having abnormal activity either very high mostly one sided activity (advertisers or fake accounts) or very low activity (deserted inactive accounts). We have grouped such clusters into common category of spammers (or fake accounts) having unusual high or low activity shown as *red nodes* in the Ego Network and all the others nodes are depicted as *blue* as shown in Figure 4 . The overall accuracy we obtained from the approach 2 was **54%.**

Overall with both approaches we saw similar types of results with few high activity nodes and rest as outliers. By comparing the results of Approach 1 and 2, we received 20-30 percent common spam accounts.

VI. Interpretation of Results

If we try to interpret the behavior of red nodes (fakeness and advertisers candidate) in ego network, we can see that some of these nodes are part of large social groups in ego network.

These can be the individuals who are a part of dense social communities (school friends, college friends, etc.), so they are also connected to many other nodes in your ego network (part of a particular social community). However, they either became totally inactive after graduation from college or they are nagging friends, which have mostly one-sided communication with the users. There can be other reasons like user joined one account that is actually running a business and advertising through social networking sites like Facebook. For example, 'House of Spices' Indian Store account is added by many Indian students at SBU. The owner uses this account for publicity and all delivery related updates almost every day. So, this account came to our result as the advertisers/spammers account.

There is another category of nodes in the result, which are outliers. They are not a part of any social group in user's ego network. These are the actual fake accounts accidentally added by users or they are actual commercial advertisers like Amazon, Dial91, etc. added by individuals according to their preferences.

VII. Conclusion

The paper presents an experimental project study to perform social interaction and behavioral analysis of friends in social network. In order to detect potential spam accounts, fake accounts or people who nag us with unsolicited one sided communication. This analysis provides an insight to us on how the friends communicate on social network and what is their frequency of communication.

In this project we collected data from Facebook accounts via history of posts, inbox messages, likes, chats, comments, tags, etc. using graph APIs for analysis. Then using this data we applied two algorithms a) Social and Interaction Behaviors and b) Markov Chain Clustering algorithm to classify accounts as spammers or fake accounts. Then we marked the accounts classified as spammers in the user's ego networks to analyze the topological positioning of spammers in our accounts. Also we validated the authenticity of our results by confirming the ground truth from respective users. We found that both the approaches gave us around 54-56% correct results i.e. users validated these persons as potential spammers (persons who nag them or very inactive accounts). When we matched results from both the approaches we found around 20-30% common results.

Overall this project gave us incredible opportunity to learn about social networks and in particular social interaction behavior in online social media like Facebook. Over the period of last two months we learnt and devised new techniques to collect data. We designed a new algorithm based upon our own experiences from using social networks. We also learnt a lot of new concepts like Markov Chain Clustering and its practical applications. This project gave us a chance to understand and solve one of major problems being faced in social networks today i.e. spamming. We think our work in this area will inspire us and other students to work on this problem and possibly create a strong impact in the field social networks.

VIII. Future Work

Since the study and analysis is done using data of ten Facebook accounts, reliability of the results is still in question. To get more reliable results, we need to analyze on large amount of dataset (at least 100 accounts). Also we want to test our algorithms on other platforms like Google+, Myspace, etc. to check their correctness.

IX. Acknowledgement

We express our sincere thanks to **Professor Jie Gao** who helped us in each and every step throughout the project progress. We faced problems and every time received genuine help and guideline from her. The novel ideas from her helped us in making our way to achieve the goals of the project.

References

[1] https://www.facebook.com/notes/k-care-shop/10-ways-to-detect-a-fake-facebook-account/ 480043898294

[2] Ahmed, F.; Abulaish, M., "An MCL-Based Approach for Spam Profile Detection in Online Social Networks," Trust, Security and Privacy in Computing and Communications (TrustCom), June 2012
[3] http://users.eecs.northwestern.edu/~kml649/publication/GaoChe12.pdf
[4] Enhua Tan, Lei Guo, Songqing Chen, Xiaodong Zhang, and Yihong Zhao. 2013. UNIK: unsupervised social network spam detection. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management (CIKM '13)
[5] https://www.cs.ucsb.edu/~chris/research/doc/acsac10_snspam.pdf