# On the Accuracy and Complexity of Spam User Detection Algorithms for Social Networks

By

Shan Xue

Submitted in partial fulfillment of the requirements for

the degree of Bachelor of Computer Science with Honours

At

Dalhousie University

Halifax, Nova Scotia

September 2019

*I would like to express my gratitude to my supervisor, Qiang Ye, for his instructive advice and useful suggestions on my thesis. I also would like to extend my deep gratefulness to my family, for their constant encouragement and support.*

## Table of Contents

# Abstract

Both Twitter and Weibo are widely-used social networking and microblogging apps with a large number of active users. However, the existence of spam users has a seriously negative economic and social impact. In this thesis, we aim to evaluate the accuracy and complexity of a series of spam user detection algorithms for social networks. Specifically, we employed two publicly-available data sets from Twitter and Weibo to quantify the performance of the algorithms under investigation. For the methods of decision trees and random decision forests, we ranked the importance of the available features, and investigated their detection accuracy and computational complexity with different sets of features. Our experimental results indicate that when only the features with high importance are adopted, the resulting detection accuracy is satisfactory while the computational complexity is seriously lower.

# Acknowledgements

First of all, I would like to express my gratitude to my supervisor, Dr. Qiang Ye for providing me the opportunity to do research at my undergraduate phase. I was directed in a clear direction and learned much. I still have far to go to achieve my research dream. Nevertheless, I am also grateful to Dr. Hai Wang, Professor in Saint Mary’s University for spending his precious time to read this thesis.

# 1. Introduction

Both Twitter and Weibo are popular online social networking and microblogging tool. Twitter has almost several hundred million active users all over the world and Weibo has built a community with same size in China. However, not all of those active users are real. Those accounts on Twitter and Weibo can be categorized to human, bots and cyborgs [1].

Sometimes, it is difficult to figure the one following you is a human or not. Bots are automated programs, which call Twitter and Weibo's APIs to perform tasks like human once they gain the login information [2]. Cyborgs are in the middle between humans and bots, which refer to either bot-assisted humans or human-assisted bots. After a person registers an account, he may set automated programs (i.e., RSS feed/blog widgets) to post tweets by calling Twitter and Weibo's APIs.

Bots and cyborgs are massively created to follow created target accounts, such as politicians, celebrities, and popular brands, to magnify their influence. Sometimes, they play follow/unfollow games to increase followers themselves, and you can find them on online accounts markets. On recent past, spammer constitute a widespread phenomenon with both economic and social impacts, which is harmful for social networks.

Unfortunately, spam user detection has been proved to be a difficult task. Recently, both Twitter and Weibo have to set some rules to limit the spam user. For instance, every Twitter account only can follow up to 400 accounts per day. As for Weibo, regardless the real number, the maximum of forward and reply numbers of a tweet shown on the homepage is one million.

Spam user detection has also attracted many attentions of researchers. Those criteria introduced earlier are mainly based on common sense and lack of validation. Machine-learning have been widely adopted in detection. However, the public datasets available are always not enough for training efficient algorithm to fit all cases. For instance, sometimes, many features used on detection are important but expensive or unavailable.

Furthermore, among the researches of detecting spam users on micro-blogging, some gave their verification only on Twitter, while some others, mainly from China, used datasets from both Twitter and Weibo. Most of them only focused on the results whether their methods can work on both of them. However, few gave deep analysis and comparison from the view of their inner differences.

## 1.1 Organization of the Thesis

The remainder of this thesis is structured as follows. Chapter 2 deals with the related work of this research area. We survey the related work about spam user detection in Twitter and micro-blogging according to several classified methods. Chapter 3 focuses on the methods and algorithms. We introduced the main algorithms that used in our experiments. These algorithms are random forest, decision tree, k-nearest neighbor, logistic regression and neural network. In Chapter 4, we describe two experiments. The first experiment generates prediction results and displays the performance of each algorithm. Then the second experiment focus on specific algorithms and apply more operations on features. Chapter 5 provides a brief summary, conclusion. Chapter 6 describes future work for this research topic.

# 2. Related Work

Seeing that very little has been done so far to give deep analysis the spam user detection in Twitter and micro-blogging, we survey the related work from the perspective of Twitter and microblogs according to several classified methods. Compared with Twitter, there are some differences in content sharing, form of expression and user behavior between microblog. Concretely, 1) Twitter's short speeches are more about reporting status, but Weibo's long speeches can become micro-media, 2) the content of micro-blog is more diverse, including video, pictures and web links adding reply, forwarding and other levels, while the content published on Twitter is relatively single, text, link is the main form, 3) microblogs are more entertaining and encourage topic discussion, while Twitter focuses more on the news itself and cannot forward comments, and 4) the media attribute of domestic microblog is strong, and the social attribute of Twitter is strong.

Initially, the research on microblog focused on social sciences such as communication and journalism. With the deepening of people's understanding, other aspects of research have emerged. The following is mainly about the spam user detection microblog and Twitter:

### 2.1 Rule-based detection

Rules have been established to detect the spam user. Two types of rules are typically used to detect spammer. First, word-based is used to identify whether a message contains a word or number of words, if it contains, it is regarded as spam [3][4]. However, the created network words such as new words and abbreviations make it difficult for word-based approaches to work on social media platforms. Second, the rules were based on pattern matching [5]. For example, if an account has been tweeting about three or more trending topics, or if an account took part in trending topics but is less than a day old, it can be classified as spammer [6]. Fire el al. [7] developed the Social Privacy Protector software for Facebook. This software scored friends for deceptiveness according to using rules based on similar relationship and chat history with others.

In addition, a well-known blogger [8] provides seven signals to recognize Twitter bots: 1) they are telling you they are a bot; 2) getting a direct response on your tweet; 3) huge amount of following and small amount of followers; 4) they tweet the same thing to everybody; 5) play the follow/unfollow game (i.e. they follow and then unfollow you within 24 hours); 6) Duplicate profile pictures, and 7) coming from an API. However, these criteria are mainly based on common sense and the authors usually do not even suggest how to validate them.

## 2.2 Content-based detection

User-generated content is the term used to describe the content contributed by web users (as opposed to the content provided by the website owners). The content generated by the user contains large amount of information that can be of utmost value for many other applications when exploited.

Content-based method focuses on the analysis of the content generated by the spam accounts. In the early network environment, the content generated by the spam accounts generally has significant identifiable characteristics, such as including prominent commercial advertisement and spam information. Initially, the user's defense is poor, and the impact of this type of spam accounts can be enormous. Therefore, the relevant research focuses on the content of the spam accounts, using the relevant technology in the field of natural language processing, through text classification [9], text sentiment analysis [10], and text orientation analysis [11], to achieve the spam follower detection.

Traditional methods of spam follower detection mainly rely on the similarity of comment content and its linguistic features to find spam commentators [12-14]. By analyzing the tendency of comment text to find the false comment [15], which is published by the spam accounts and deviates from the normal user's comment. In addition, the characteristics of the normal user comment mode [57] presented by false comments are used to identify the network water forces hidden behind these comments, such as repeated use of a large number of unsubstantiated adjectives, multiple repetitions of language, etc. By analyzing the tendency of comment text to mine anomaly comment, which involves certain natural language processing, its efficiency is low. The statistical model that pays attention to a large number of anomaly comment avoids the bottleneck of natural language processing, and uses statistical theory to search for anomaly comment, its efficiency and accuracy are higher. In [16] verifies that the conclusion of false comments can be well recognized by using the text features of comments themselves.

However, traditional spam follower detection methods can only find a certain type of spam accounts. With the enhancement of user's discrimination, the spam account has limited impact only by creating a large number of similar spam comments. Therefore, the spammer shows a more normal user behavior, and its deception strategies are gradually diversified. The recognition effect of the new spam account based on content feature recognition method is declining.

### 2.3 URL-based detection

For the detection of URLs in the content of the information, it is usually established by creating a blacklist. According to the hyperlink contained in the Weibo content and the list of some existing offensive web pages (Blacklist), determine whether the Weibo user is a spam user. For example, H.Gao. et al [17] analyzes the URLs embedded in the content on Facebook to detect spam on Facebook. K. Thomas [18] designed a Monarch system to determine the water user by analyzing the content of Twitter and detecting the URL and the page of the jump.

In addition to hyperlinks in the content detection, the spam disseminated by the spam account may be directly in the Weibo. A large number of studies have shown that there are generally more obvious keywords in the blog post information of the advertising, such as "@", "##" and some referrals, etc., or the content from spam account will be repetitive. In particular, the spam account edits content and publishes information through some kind of software [19]. In response to this feature, some researchers identified spam accounts by analyzing the differences between keywords in spam accounts terms and common terms.

In Weibo, users will carry personal information, content detection is not limited to the content of the forwarding, but also the user's attribute information, that is, personal data and labels. This information includes user ID, gender, number of followers, number of fans, number of forwards, number of comments, etc., among which it is currently used to identify the spam account by observing the fan ratio. In order to expand its influence, spam account pays a lot of attention to normal users in order to achieve the goal of "mutual fans" in Weibo. As a result, the spam account has different characteristics from normal users in terms of fans ratio. In addition to focusing on the visual features of fans, user names with specific rules or characters, the growth rate of number of attentions, the proportion of active users in attention and so on can also be used to identify features in the spam account.

### 2.4 User behavior-based detection

Information dissemination has different characteristics from traditional network in Weibo. Meanwhile, with the increasing application of machine learning, the spammer detection has changed from content and link detection to user characteristics and user behavior analysis. Specifically, in addition to publishing the content itself, the user information may also transmit information through operations such as forwarding, attention, and private information. In terms of behavior, in order to attract more attention, the spam account will have some behaviors different from normal users, such as a lot of attention to other users, mutual accounts between spam accounts, frequent posting of Weibo, etc., which have produced the difference between the spam account and the normal user.

There are many research methods based on the characteristics of user behavior. In the microblog marketing, in order to achieve the purpose of influencing or changing the user's purchase decision, the spammer will give a lot of praise to the target goods to encourage users to purchase or a large number of bad reviews to destroy the competitive goods. Lim et al. [20] summarized several representative spam account behaviors by analyzing a large number of user comments in Amazon, such as high repetitiveness of comments and sudden bursts of time. Mukherjee et al. [21] constructed a Bayesian recognition model based on the suspiciousness of the spam account. According to user statistics in the YouTube website [22], these data confirm the existence of a large number of spam accounts in social networks, and they also define spam video in online video sharing sites. They use manual tagging methods to build training data sets, then analyze the behavior of identified the spam account, and use machine learning algorithms to classify users. Lin et al. [23] uses the "honeypot technology" in network security combined with machine learning to collect the spam accounts in Weibo. However, regardless of the technology, it is limited by the size and collection lag of the spam information data set. The application of the blacklist-based detection mechanism in the Weibo network needs further research.

With the upgrade of the spammers' hidden strategy, its behavior is more and more concealed and tends to normal users. The traditional content-based and behavior-based identification methods are also facing challenges, and the bottleneck is increasingly prominent. On the one hand, the spammer detection method based on comprehensive features can achieve higher accuracy and recall rate. On the other hand, in-depth exploration of essential characteristics of spam accounts, defines high-discrimination characteristics and behavior patterns will become the key to the detection of the spammer.

### 2.5 User relationship-based detection

In social networks, users gradually form a user-centered social circle through interaction, and the social relationship among users contains rich user information. Compared with normal users, the spam account in social networks does not have normal social relations, and its formed network structure is special. For example, the spammer generally has a large number of extremely unbalanced fans. Therefore, the user relationship characteristics in the social network are very distinguishable when identifying the spam account in the field of social network.

Song et al. [24] considered that the spammer detection method based on user behavior characteristics has certain hysteresis and easy forgery, that is, after the user has performed the behavior similar to the spam account, the method can identify it, and the user characteristics it is easy to be masked by the spam account. But the whole network relationship has certain stability, and its characteristics are not easily affected by user behavior. Therefore, using this type of feature has a better effect on the spammer detection. Murmann [25] uses the neighbor nodes with direct interaction to detect the trust relationship between users and obtain a new relationship feature set. Using this feature set to sort the users suspiciously, the most suspicious one is the spam account. Moh et al. [26] also uses the user social relationship in Twitter, such as its friends and fan characteristics, to obtain the trustworthiness of the user through different feature matrices, to determine whether the user is a spammer.

In addition, Gayo-Avello et al. [27] used the network water army in Twitter to spend a lot of time to become a fan to target users or wait for the target users to be became a fan to discover the spam account, and proposed the topic ranking of the most concerned network in the network. They also proposed to use user influence characteristics to improve the accuracy of the spammer detection. Krestel et al. [28] used the characteristics of spammer suspiciousness to spread in social networks, using the propagation model to discover tags on the graph model. This method realizes the modeling of the relationship structure between the spam account, tags and network resources in the tag sharing site. Bhat et al. [29] found that, similar to normal users, the spam account in the social field can also form a certain degree of spam community. Therefore, Bhat et al. extracted user interaction diagrams from user behavior logs and found overlapping community maps. After labeling some of the spam account nodes manually, each node to be identified was calculated. The community relationship with the marked nodes is used to classify the unknown nodes.

## 2.6 Environment-based detection

The environmental characteristics of the spam accounts are mainly in the network environment level, and the spam account will be different from the normal users. The spammer detection research based on environmental characteristics is based on the environmental characteristics generated by the spam account when it conducts harmful behaviors. The environmental characteristics cannot be modified by the spam account, so its recognition accuracy is high. However, most of the spammer detection research based on environmental characteristics requires corresponding experimental data sets, so its popularization is lower than other research methods.

Ramachandran et al. [30] used the IP-based blacklist information, TCP footprint information, routing information, and robot website command tracking information for the spammer detection. Schatzmann et al. [31] believe that the identification of the behavioral characteristics based on the level of the network protocol cannot meet the needs of the spammer detection. They propose to analyze the spammer behavior from the core part of the network to realize the detection of complex spammer. They use A national ISP (Internet service provider) network spammer behavior record, from the ISP perspective to propose traffic level characteristics, to achieve the modeling of spammer behavior. Xu et al. [32] proposed a method based on the identification pattern of the spammer resources. It is found that the spammer always uses open resources that can be easily obtained. Therefore, according to the sudden use mode of resources, it can be very good to identify the spam account.

Additionally, Las-Casas et al. [33] proposed a method for identifying the source of the spammer, that is, based on the network feature recognition when the spam account was generated, and using the data records of the Brazilian broadband ISP as the experimental data set. They proposed that it is difficult to accurately discover the spam account based on the behavior of a single network, but it is easy to discover the characteristics from the perspective of network traffic. Because the spam account minimizes the workload and maximizes the benefits, it will concentrate. In a period of time, a large amount of garbage is produced, so the network load will suddenly increase during this period, and the traffic will be concentrated in some links. Compared with the traditional spammer detection method, the accuracy of this method is higher. However, this method requires the application of ISP data and cannot be universally promoted.

### 2.7 Hybrid-based detection

The researches of the spammer detection described above are based on the specific types of spam account characteristics, but these methods cannot comprehensively analyze the spammer's behavior, so its recognition accuracy has a bottleneck. On this basis, the integration of a variety of specific types of methods for each target area of the spam account has a higher detection accuracy.

Traditional spammer detection method judges whether it is a spam account or not according to the content generated by users. Therefore, it is common to use integrated feature method based on user behavior, relationship and published content. For example, Zinman et al. [34] used naive Bayesian and neural network methods to model users in social networks, and divided them into four types according to their activity. This paper analyzed the characteristics of user behavior and relationship in social networks, identified them according to the salient behavior patterns of spam account, and broadened the classification of users in social networks. Benevenuto et al. [35] divides the spammers in online video sharing sites into two fine-grained kinds according to their purposes, and gives the definitions separately. In addition, this method establishes a YouTube user tag data set using manual tagging, which provides a test data set for the detection of this type of spam accounts.

In Twitter, according to the aim of maximizing the impact of the spammer, Benevenuto et al. [36] analyzed the three most popular topics on Twitter, tagged the users involved, and used users' Tweets and their behavior characteristics to judge whether they are spam accounts. However, the experimental data used in this method are only part of the users who participate in the hot topics, and the coverage of users is limited, so the learning effect is limited. The recognition accuracy of the spam account is only 70%, but the detection accuracy of the normal users can reach 96%. Amleshwaram et al. [37] synthesized the characteristics of users in various aspects of social networks, such as behavior, content, user relationships, and so on. They realized the rapid detection of spam accounts in the social field, and the time and resources needed for identification were greatly reduced. They conducted a cluster analysis of the unknown spam account and found some spammer groups popular on Twitter. At the same time, they found that most of the spam accounts on Twitter had very few Tweets, and their main goal was to spread spam and create network influences.

In Weibo, Lin et al. [38] used a variety of channels to collect a large number of spam accounts from Sina Weibo, including more than 1,000 accounts collected by “traps” and web crawlers using keyword search and manual tagging. Collected spam accounts directly purchased 8600 Sina Weibo zombie fans. According to their different purposes, the spammers are divided into three types, which analyze their behavioral relationship characteristics and construct a specific type of spam account classifier. They systematically divide the spammers according to its behavior characteristics, and construct corresponding classifier in order to achieve higher accuracy. At the same time, it covers most of the spam accounts that may exist in Sina Weibo because of the data widely available. Therefore, the data set has a good performance in spammer detection in Sina Weibo network.

# 3 Algorithms for Spam User Detection

## 3.1 Random Forest

Random forests are an ensemble learning method used for classification and regression, which actually consists of many decision trees. It creates as many trees on the subset of training data and combines the output of all the trees. In this way it reduces overfitting problem in decision trees and also reduces the variance and therefore improves the accuracy. The disadvantage of random forest is that a large number of trees can make the algorithm too slow and ineffective for real-time predictions.

Random forests use a modified tree learning algorithm that selects, at each candidate split in the learning process, a random subset of the features. This process is sometimes called "feature bagging". The reason for doing this is the correlation of the trees in an ordinary bootstrap sample: if one or a few features are very strong predictors for the response variable (target output), these features will be selected in many of the B trees, causing them to become correlated.

## 3.2 Decision Tree

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. DTs have some advantages, for example, simple to understand and to interpret, requires little data preparation and able to handle multi-output problems. The disadvantage is that over-matching may occur.

A decision tree is a predictive model which is a mapping from observations about an item to conclusions about its target value. In the tree structures, leaves represent classifications (also referred to as labels), non-leaf nodes are features, and branches represent conjunctions of features that lead to the classifications. Especially, the C4.5 and C5.0 algorithms are forms of decision trees and are among the most popular among classification algorithms.

## 3.3 K-NN

Generally, the k-nearest neighbor algorithm (k-NN) is a non-parametric method used for classification and regression according to measuring the distance between different features. It is easy to achieve high precision, insensitivity to outliers, and no input assumptions. The disadvantages are high computational complexity and high space complexity.

The way it works is described as follows: given a sample data set, also called a training sample set, and each data in the sample set has a label, that is, we know the correspondence between each data in the sample set and the belonging category. After inputting the new data without the label, compare each feature of the new data with the feature corresponding to the data in the sample set, and then extract the classification label of the most similar data (nearest neighbor) of the feature in the sample set. In general, we only select the top k most similar data in the sample data set, usually k is an integer no larger than 20. Finally, select the category with the most occurrences among the k most similar data as the classification of the new data.

## 3.4 Logistic Regression

Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, and so on. The advantage is that the computational cost is not high, and it is easy to understand and implement. The disadvantage is that it is easy to underfit and the classification accuracy may not be high.

In general, logistic regression classifier utilizes the logistic function or sigmoid function to model a binary dependent variable. Specifically, it can use a linear combination of more than one feature value or explanatory variable as argument of the sigmoid function. The corresponding output of the sigmoid function is a number between 0 and 1. The middle value is considered as threshold to establish what belong to the class 1 and to the class 0. In particular, an input producing an outcome greater than 0.5 is considered belong to the class 1. Conversely, if the output is less than 0.5, then the corresponding input is classified as belonging to 0 class.

### 3.5 Neural Network

Neural networks are simple models of the way the nervous system operates. The basic processing units are arranged in layers. There are typically three parts in a neural network: an input layer, with units representing the input fields; one or more hidden layers; and an output layer, with a unit or units representing the target fields. The units are connected with varying connection strengths (or weights). Input data are presented to the first layer, and values are propagated from each neuron to every neuron in the next layer. Eventually, a result is delivered from the output layer.

The network learns by examining individual records, generating a prediction for each record, and adjusting the weights whenever it makes an incorrect prediction. This process is repeated many times, and the network continues to improve its predictions until one or more of the stopping criteria have been met. After the training progresses, the network can be applied to future cases where the outcome is unknown.

# 4 Experimental Results

In the experimental section, we applied two experiments based on the collected data sets. The first experiment quantifies the performance of varied algorithms with Twitter and Weibo data sets. Each algorithm generates their prediction metrics which include accuracy, precision, recall, and F-measure. The second experiment focus on low-weight features that generated by specific algorithms. The feature weights can be computed when Random Forest or Decision Tree algorithm is applied. Then a proper threshold can be chosen based on feature weights, and that will lead to different complexity and prediction results.

## 4.1 Evaluation methodology

The prediction metrics used in the experiments are accuracy, precision, recall, and F-measure. Table 4.1 below gives the confusion matrix that used to calculate the prediction results. Each column represents the actual values, while each row represents the predicted values.

| | | Actual Values | |
|---|---|---|---|
| | | Positive | Negative |
| Predicted Values | Positive | True Positive (TP) | False Positive (FP) |
| | Negative | False Negative (FN) | True Negative (TN) |

**Table 4.1** Confusion matrix.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.2}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.3}$$

$$f_1\text{-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{4.4}$$

In order to evaluate each algorithm, the below prediction metrics will be considered.

Accuracy: the proportion of predicted true results (both true positives and true negatives) in the population.

Precision: the proportion of predicted positive cases that are indeed real positive.

Recall: the proportion of real positive cases that are indeed predicted positive.

F-measure (f1-score): the harmonic mean of precision and recall.

## 4.2 Feature weight/importance

In the second experiment, we will focus on feature importance in order to remove low-weight features. The feature weight is the importance of a feature. The higher weights, the more important the feature. Feature importance is computed as the total reduction of the criterion brought by the that feature. The formula below is defined to compute the weight/importance of a feature.

$$N_t/N * (impurity - N_{tR}/N_t * right\ impurity - N_{tL}/N_t * left\ impurity)$$

(4.5)

*N* is the total number of samples, *Nt* is the number of samples at the current node, impurity is the gini/entropy value, NtR is the number of samples in the right child. NtL is the number of samples in the left child.

## 4.3 Algorithm performance

In this subsection, we applied algorithms include Random Forest, Decision Tree, K-NN, Logistic Regression, and Neural Network. There are three sets of performance, each set only test one data set with specific features. The first set focus on the Weibo data sets and applied algorithms with 11 features. The second set focus on the Twitter data sets with 23 features. The third set focus on the Twitter data sets with 11 specific features which are the same features used in the first set. Table 4.2 summarizes the results by the application of each algorithm.

| | Algorithm | Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|---|
| Weibo datasets with 11 features | Random forest | 0.9229 | 0.5417 | 0.5778 | 0.5591 |
| | Decision tree | 0.9154 | 0.50 | 0.5333 | 0.5161 |
| | k-Nearest Neighbors | 0.9361 | 0.6486 | 0.5333 | 0.5854 |
| | Logistic Regression | 0.9211 | 0.5319 | 0.5556 | 0.5435 |
| | Neural Network | 0.9135 | 0.0 | 0.0 | Nan |
| | | | | | |
| Twitter datasets with 23 features | Random forest | 0.9837 | 0.9930 | 0.9811 | 0.9870 |
| | Decision tree | 0.9742 | 0.9820 | 0.9772 | 0.9796 |
| | k-Nearest Neighbors | 0.8102 | 0.8593 | 0.8371 | 0.8481 |
| | Logistic Regression | 0.6329 | 0.6329 | 1.0 | 0.7752 |
| | Neural Network | 0.6329 | 0.6329 | 1.0 | 0.7752 |
| | | | | | |
| Twitter datasets with 11 features | Random forest | 0.9598 | 0.9692 | 0.9672 | 0.9682 |
| | Decision tree | 0.9604 | 0.9711 | 0.9662 | 0.9686 |
| | k-Nearest Neighbors | 0.9623 | 0.9721 | 0.9682 | 0.9701 |
| | Logistic Regression | 0.9390 | 0.9749 | 0.9275 | 0.9506 |
| | Neural Network | 0.9705 | 0.980 | 0.9732 | 0.9766 |

**Table 4.2 Experimental Results**

## 4.4 Threshold for feature weight/importance

The section 4.2 above mention that Random Forest and Decision Tree algorithms are able to compute the feature importance. Table 4.3 below summarizes each feature with its importance value with 11 features in Weibo dataset. Figure 4.1 and Figure 4.2 present the feature importance on bar charts. Then Table 4.4, Figure 4.3 and Figure 4.4 present the feature importance with 23 features in Twitter dataset. In the same way, Table 4.5, Figure 4.5 and Figure 4.6 present the records in Twitter dataset but with 11 features.

|  | Feature importance |  |
|---|---|---|
| **Weibo dataset with 11 features** | **Random forest** | **Decision tree** |
| ratio_friends_followers_square | 0.630755 | 0.462782 |
| followers_2_times_ge_friends | 0.156805 | 0.320674 |
| has_image | 0.091240 | 0.144926 |
| nb_tweets_ge_50 | 0.055567 | 0.032456 |
| friends_ge_100 | 0.026267 | 0.021980 |
| followers_ge_30 | 0.025508 | 0.015709 |
| no_tweets | 0.007084 | 0.001473 |
| ratio_friends_followers_ge_50 | 0.006701 | 0.000000 |
| nb_friends | 0.000074 | 0.000000 |
| ratio_friends_followers_around_100 | 0.000000 | 0.000000 |
| has_name | 0.000000 | 0.000000 |

**Table 4.3 Weibo Data Set: Importance of Features**

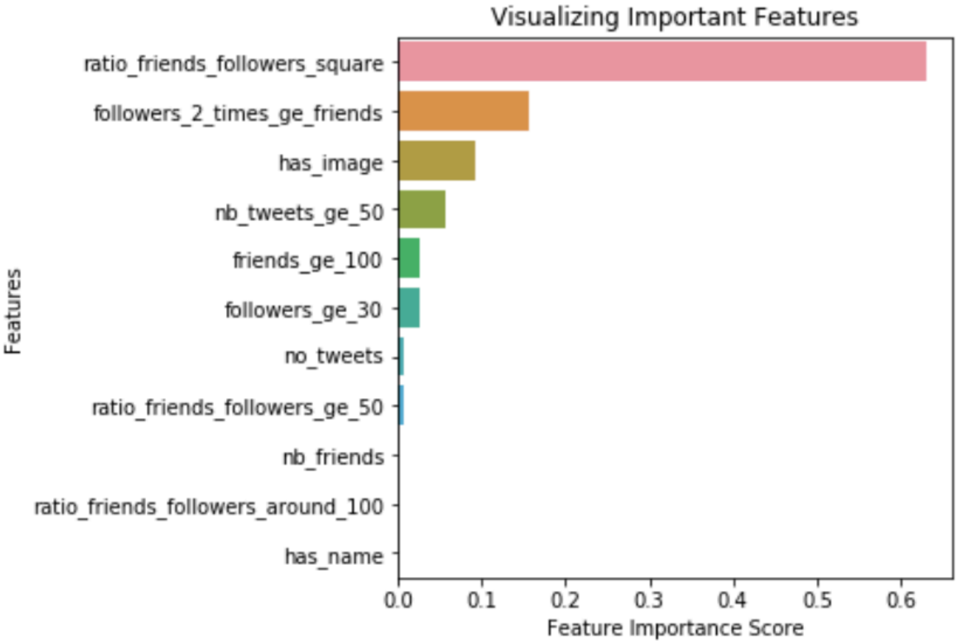

**Figure 4.1 Random forest feature importance, Weibo dataset with 11 features**

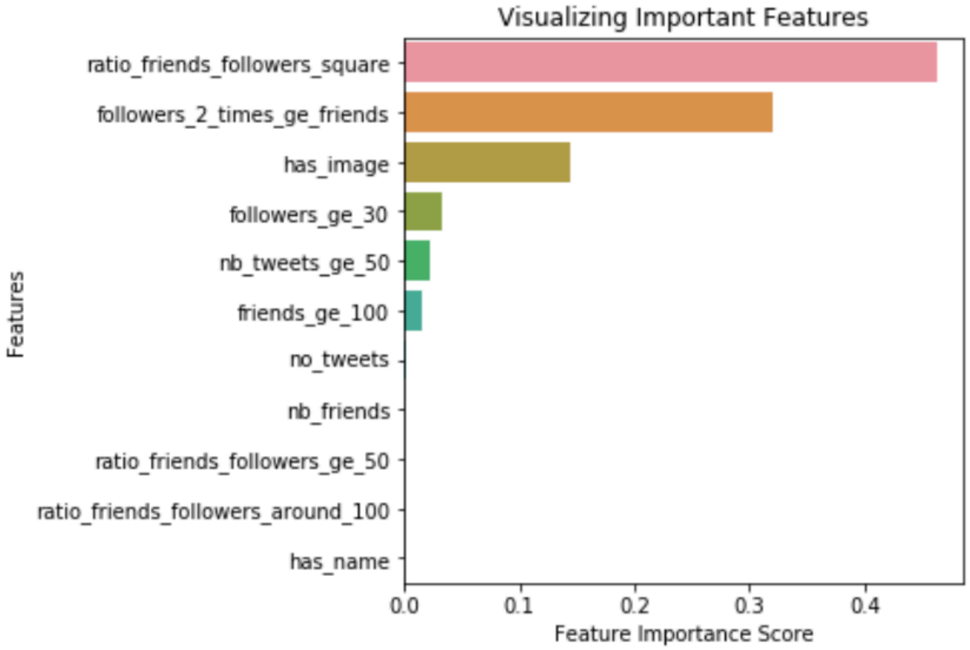

**Figure 4.2 Decision tree feature importance, Weibo dataset with 11 features**

| | **Feature importance** | |
|---|---|---|
| **Twitter dataset with 23 features** | **Random forest** | **Decision tree** |
| ratio_friends_followers_square | 0.359056 | 0.729877 |
| followers_ge_30 | 0.160349 | 0.069252 |
| age | 0.104237 | 0.065958 |
| belongs_to_a_list | 0.063486 | 0.061181 |
| followers_2_times_ge_friends | 0.054842 | 0.026673 |
| following_rate | 0.047866 | 0.020853 |
| url_in_profile | 0.045469 | 0.009565 |
| friends_ge_100 | 0.033544 | 0.005806 |
| has_biography | 0.026018 | 0.005352 |
| nb_tweets | 0.024904 | 0.004859 |
| no_tweets | 0.020470 | 0.000623 |
| nb_tweets_ge_50 | 0.014345 | 0.000000 |
| no_bio | 0.013622 | 0.000000 |
| ratio_friends_followers_ge_50 | 0.012443 | 0.000000 |
| has_address | 0.007835 | 0.000000 |
| no_location | 0.004614 | 0.000000 |
| ratio_friends_followers_around_100 | 0.001960 | 0.000000 |
| duplicate_profile_picture | 0.001537 | 0.000000 |
| default_image_after_2_month | 0.001341 | 0.000000 |
| has_image | 0.001027 | 0.000000 |
| nb_friends | 0.001019 | 0.000000 |
| bot_in_biography | 0.000017 | 0.000000 |
| has_name | 0.000000 | 0.000000 |

**Table 4.4 Twitter Data Set with 23 Features: Importance of Features**

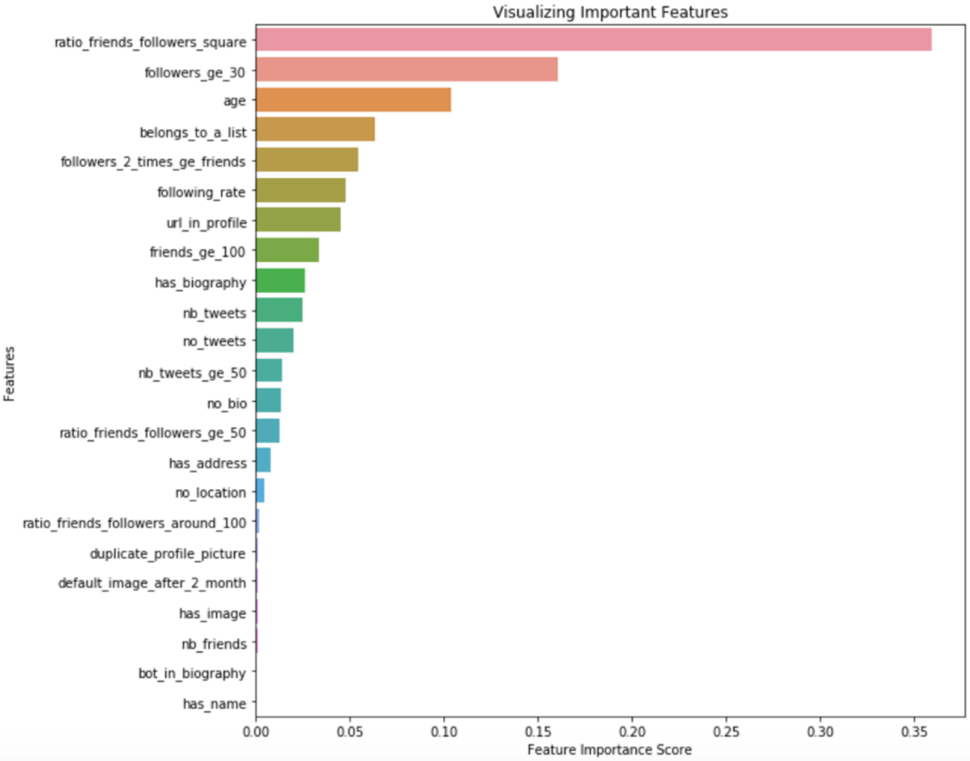

Figure 4.3 Random forest feature importance, Twitter dataset with 23 features

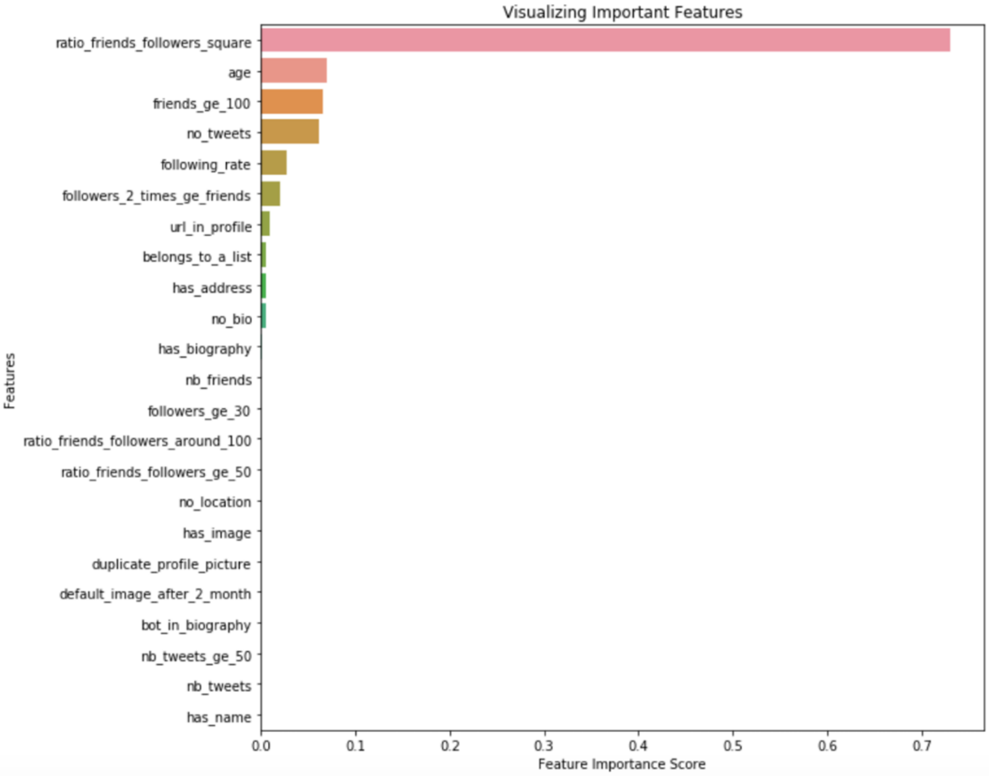

Figure 4.4 Decision tree feature importance, Twitter dataset with 23 features

| | Feature importance | |
|---|---|---|
| **Twitter dataset with 11 features** | **Random forest** | **Decision tree** |
| ratio_friends_followers_square | 0.516891 | 0.822388 |
| followers_ge_30 | 0.236034 | 0.067979 |
| followers_2_times_ge_friends | 0.101012 | 0.067897 |
| friends_ge_100 | 0.058848 | 0.021731 |
| no_tweets | 0.047212 | 0.011375 |
| nb_tweets_ge_50 | 0.026647 | 0.005519 |
| has_image | 0.005541 | 0.002050 |
| ratio_friends_followers_ge_50 | 0.003980 | 0.001061 |
| nb_friends | 0.001958 | 0.000000 |
| ratio_friends_followers_around_100 | 0.001876 | 0.000000 |
| has_name | 0.000000 | 0.000000 |

**Table 4.5 Twitter Data Set with 11 Features: Importance of Features**

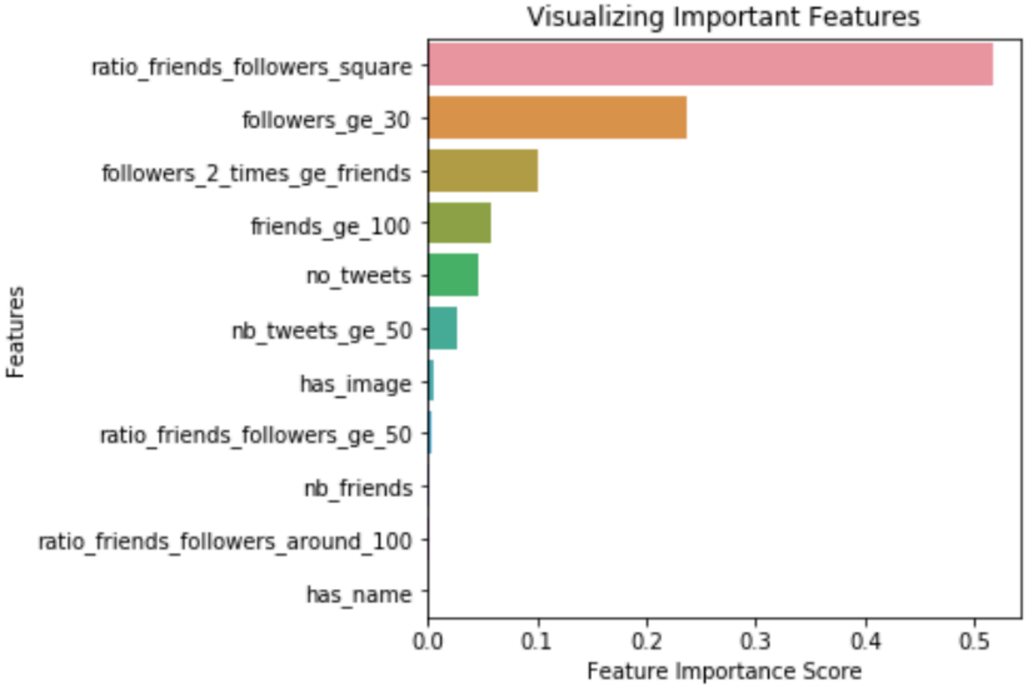

**Figure 4.5 Random forest feature importance, Twitter dataset with 11 features**

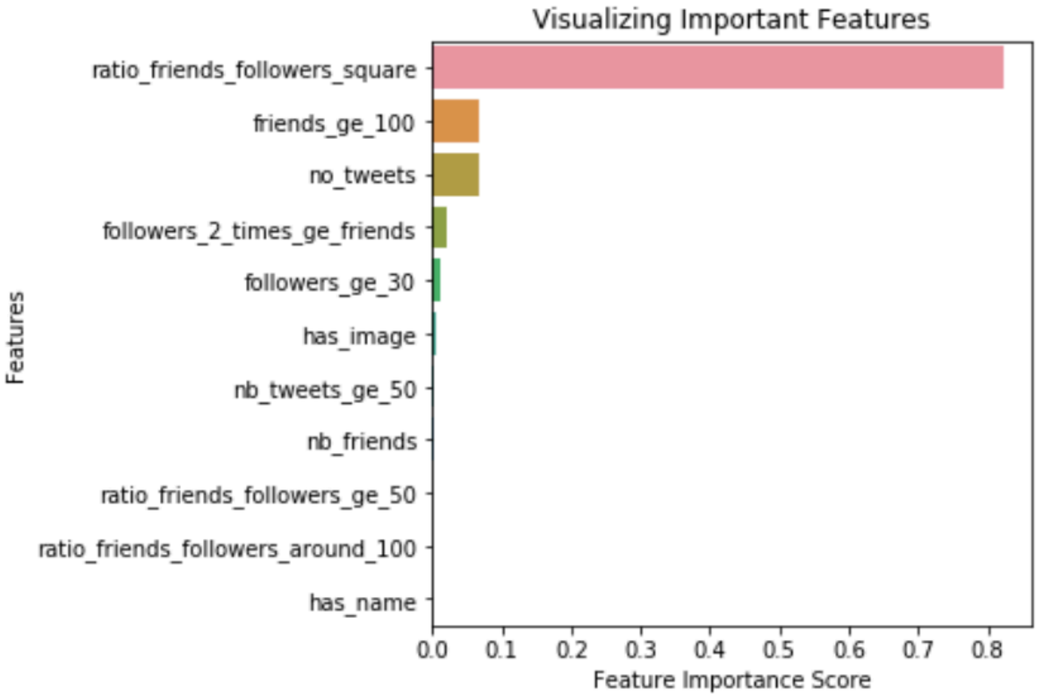

**Figure 4.6 Decision tree feature importance, Twitter dataset with 11 features**

By observation the tables, a proper threshold can be chosen which threshold values is 0.03. The feature weight/importance which less than 0.03 will be deleted.

Table 4.6 shows the prediction results of random forest after deleting low-weight features in Twitter data sets. We focus on accuracy value between Random Forest and Decision Tree algorithms. The accuracy is around 0.9604 which is close to original accuracy 0.9598 after removing low-weight features. For decision tree, Table 4.6 shows its accuracy result after removing low-weight features is around 0.9573 which is a slight less than original precision 0.9604. Similarly, the accuracy values have small differences in Weibo dataset. For Random Forest, the accuracy is decreased from 0.9229 to 0.9060. For Decision Tree, the accuracy is decreased from 0.9154 to 0.8929.

|  | Algorithm | Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|---|
| Weibo datasets with 11 features | Random forest | 0.9060 | 0.4510 | 0.5111 | 0.4792 |
|  | Decision tree | 0.8929 | 0.3571 | 0.3333 | 0.3448 |
|  |  |  |  |  |  |
| Twitter datasets with 11 features | Random forest | 0.9604 | 0.9701 | 0.9672 | 0.9687 |
|  | Decision tree | 0.9573 | 0.9662 | 0.9662 | 0.9662 |

**Table 4.6 Experimental Results**

The time complexity of decision tree is *O(Nkd)*. *N* is the number of training examples, *k* is the number of features, and *d* is the depth of the decision tree. Thus, the time complexity will decrease after removing the number of features. For random forest, it is an ensemble model of decision trees. It creates as many trees on the subset of training data and combines the output of all the trees. Therefore, the time complexity of random forest will decrease by reducing the number of features. In our experiments, the collected data sets in experiment are not large enough, the runtime cannot be test and have a satisfactory result.

# 5 Conclusion

In this paper, we proposed to evaluate the accuracy and complexity of a series of spam user detection algorithms for social networks. The prediction metrics include accuracy, precision, recall and F-measure. In the case with more features, Random Forest and Decision Tree have high prediction results. When less features used, all of the algorithms supply high prediction results. For Random Forest and Decision Tree algorithms, we ranked the importance of the available features, then remove low importance features based on a threshold. After removing features, the new accuracy has slightly differences with original accuracy. According to our experimental results, the time complexity of Random Forest and Decision Tree tends to be lower by reducing low importance features. It indicates that when only the feature with high importance are adopted, the resulting detection accuracy is satisfactory while the complexity is lower.

# 6 Future Work

The future work involves adding more features. The features are built from attributes of data. For example, the semantic analysis could be used to determine the gender of users by analyzing their posts or tweets. Then compare with the gender attribute from user profile. However, the important step to do social media analysis is to collect clearly and large number of data. In fact, it is not easy to collect data due to company license and restriction of sharing source data. Some company supply the API for developer to process data such as Twitter but with strict restrictions. Thus, data collection needs to spend a long term before any appropriate ways exist.

## Bibliography

[1] Chu, Zi & Gianvecchio, Steven & Wang, Haining & Jajodia, Sushil. (2012). Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg?. Dependable and Secure Computing, IEEE Transactions on. 9. 811-824. 10.1109/TDSC.2012.75.

[2] J. Oentaryo, Richard & Murdopo, Arinto & Prasetyo, Philips Kokoh & Lim, Ee-Peng. (2016). On Profiling Bots in Social Media. 92-109. 10.1007/978-3-319-47880-7_6.

[3] M. Jiang, P. Cui, and C. Faloutsos, Suspicious behavior detection: Current trends and future directions, IEEE Intell. Syst., vol. 31, no. 1, pp. 31_39, Jan. 2016.

[4] P. Hayati and V. Potdar, Toward spam 2.0: An evaluation of Web 2.0 anti-spam methods, in Proc. 7th IEEE Int. Conf. Ind. Inform. (INDIN), Jun. 2009, pp. 875_880.

[5] S. Gurajala, J. S. White, B. Hudson, B. R. Voter, and J. N. Matthews, Pro_le characteristics of fake Twitter accounts, Big Data Soc., vol. 3, no. 2, p. 2053951716674236, 2016.

[6] H. Kwak, C. Lee, H. Park, and S. Moon, What is Twitter, a social network or a news media? in Proc. 19th Int. Conf.WorldWideWeb, 2010, pp. 591_600.

[7] M. Fire, D. Kagan, A. Elyashar, andY. Elovici, Friend or foe? Fake profile identification in online social networks, Social Netw. Anal. Mining, vol. 4, no. 1, p. 194, 2014.

[8] Stateofsearch.com, How to Recognize Twitterbots: 7 Signals to Look out for, https://www.stateofdigital.com/how-to-recognize-Twitter-bots-6-signals-to-look-out-for/August 2012. (Last checked 23/09/19)

[9] Sriram B, Fuhry D, Demir E, Ferhatosmanoglu H, Demirbas M. Short text classification in Twitter to improve information filtering. In: Crestani F, Marchand-Maillet S, Chen HH, eds. Proc. of the 33rd Int’l ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR 2010). New York: ACM Press, 2010. 841−842.

[10] Zhao YY, Qin B, Liu T. Sentiment analysis. Ruan Jian Xue Bao/Journal of Software, 2010,21(8):1834−1848.

[11] Duh A, Stiglic G, Korosak D. Enhancing identification of opinion spammer groups. In: Proc. of the Int’l Conf. on Making Sense of Converging Media (AcademicMindTrek 2013). New York: ACM Press, 2013. 326−328.

[12] Lau RYK, Liao SY, Kwok RCW, Xu K, Xia Y, Li Y. Text mining and probabilistic language modeling for online review spam detection. ACM Trans. on Management Information Systems (TMIS), 2011,2(4):25.

[13] Li F, Huang M, Yang Y, Zhu X. Learning to identify review spam. In: Walsh T, ed. Proc. of the 22nd Int’l Joint Conf. on Artificial Intelligence (IJCAI 2011), Vol.3. Menlo Park: AAAI Press, 2011. 2488−2493. [doi: 10.5591/978-1-57735-516-8/IJCAI11-414]

[14] Liu HY, Zhao YY, Qin B, Liu T. Comment target extraction and sentiment classification. Journal of Chinese Information Processing, 2010,24(1):84−88 (in Chinese with English abstract).

[15] Jindal N, Liu B, Lim EP. Finding unusual review patterns using unexpected rules. In: Huang J, Koudas N, Jones G, eds. Proc. of the 19th ACM Int’l Conf. on Information and Knowledge Management (CIKM 2010). New York: ACM Press, 2010. 1549−1552.

[16] Ott M, Choi Y, Cardie C, Hancock JT. Finding deceptive opinion spam by any stretch of the imagination. In: Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol.1. Stroudsburg: ACL, 2011.309−319.

[17] Indyk W, Kajdanowicz T, Kazienko P, et al. Web Spam Detection Using MapReduce Approach to Collective Classification[J]. Advances in Intelligent Systems & Computing, 2013, 189:197-206.

[18] Zhang L, Zhu J, Yao T. An evaluation of statistical spam filtering techniques[J]. ACM Transactions on Asian Language Information Processing, 2004, 3(4):243-269.

[19] Hongyu L, Yanyan Z, Bing Q, et al. Comment Target Extraction and Sentiment Classification[J]. Journal of Chinese Information Processing, 2010.

[20] Lim E P, Nguyen V A, Jindal N, et al. Detecting product review spammers using rating behaviors[C]// Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010. ACM, 2010.

[21] Mukherjee A, Kumar A, Liu B, et al. Spotting opinion spammers using behavioral footprints[C]// Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013.

[22] Benevenuto, Fabrício, Rodrigues T , Almeida, Virgílio A. F, et al. Detecting spammers and content promoters in online video social networks [J]. 2009.

[23] Lin C, He J, Zhou Y, et al. [ACM Press the 7th Workshop - Chicago, Illinois (2013.08.11-2013.08.11)] Proceedings of the 7th Workshop on Social Network Mining and Analysis - SNAKDD \"13 - Analysis and identification of spamming behaviors in Sina Weibo microblog[J]. 2013:1-9.

[24] Song J, Lee S, Kim J. Spam Filtering in Twitter Using Sender-Receiver Relationship[C]// Recent Advances in Intrusion Detection-international Symposium. DBLP, 2011.

[25] Murmann AJ. Enhancing spammer detection in online social networks with trust-based metrics [MS. Thesis]. San Jose: San Jose State University, 2009.

[26] Moh TS, Murmann AJ. Can you judge a man by his friends? Enhancing spammer detection on the Twitter microblogging platform using friends and followers. In: Prasad SK, Vin HM, Sahni S, eds. Proc. of the Int'l Conf. on Information Systems and Technology Management (ICISTM 2010). Heidelberg: Springer-Verlag, 2010. 210−220.

[27] Gayo-Avello D, Brenes DJ. Overcoming spammers in Twitter—A tale of five algorithms. In: Proc. of the Spanish Conf. on Information Retrieval (CERI 2010). 2010. 41−52.

[28] Krestel R, Chen L. Using co-occurrence of tags and resources to identify spammers. In: Saeys Y, Liu H, Inza I, eds. Proc. of the Discovery Challenge Workshop at the European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2008). Brookline: Microtome Publishing, 2008. 38−46.

[29] Bhat SY, Abulaish M. Community-Based features for identifying spammers in online

social networks. In: Rokne JG, Faloutsos C, eds. Proc. of the 2013 IEEE/ACM Int’l Conf. on Advances in Social Networks Analysis and Mining (ASONAM 2013). New York: ACM Press, 2013. 100−107.

[30] Ramachandran A, Feamster N. Understanding the network-level behavior of spammers. New York: ACM Press, 2006, 291-302.

[31] Schatzmann D, Burkhart M, Spyropoulos T. Inferring spammers in the network core. In: Moon SB, Teixeira R, Uhlig S, eds. Proc. of the 10th Int’l Conf. on Passive and Active Network Measurement (PAM 2009). Heidelberg: Springer-Verlag, 2009. 229−238.

[32] Xu KS, Kliger M, Hero III AO. Identifying spammers by their resource usage patterns. In: Proc. of the 7th Annual Collaboration Electronic Messaging, Anti-Abuse and Spam Conf. (CEAS 2010). 2010.

[33] Las-Casas PHB, Guedes D, Almeida JM, Ziviani A, Marques-Neto HT. SpaDeS: Detecting spammers at the source network. Computer Networks, 2012,57(2):526−539.

[34] Zinman A, Donath J. Is britney spears spam. In: Proc. of the 4th Conf. on Email and Anti-Spam (CEAS 2007). 2007. 1−10. http://ceas.cc/2007/

[35] Benevenuto F, Rodrigues T, Almeida V, Almeida J, Goncalves M. Detecting spammers and content promoters in online video social networks. In: Allan J, Aslam J, Sanderson M, Zhai C, Zobel J, eds. Proc. of the 32nd Int’l ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR 2009). New York: ACM Press, 2009. 620−627.

[36] Benevenuto F, Magno G, Rodrigues T, Almeida V. Detecting spammers on Twitter. In: Proc. of the 7th Annual Collaboration Electronic Messaging, Anti-Abuse and Spam Conf. (CEAS 2010), Vol.6. 2010. 12−20.

[37] Amleshwaram AA, Reddy N, Yadav S, Gu G, Yang C. CATS: Characterizing automation of Twitter spammers. In: Proc. of the 5th Int’l Conf. on Communication Systems and Networks (COMSNETS 2013). Washington: IEEE Computer Society, 2013. 1−10.

[38] Lin C, Zhou Y, Chen K, He J, Yang X, Song L. Analysis and identification of spamming behaviors in Sina Weibo microblog. In: Zhu F, He Q, Yan R, eds. Proc. of the 7th Workshop on Social Network Mining and Analysis (SNAKDD 2013). New York: ACM Press, 2013. 5−13.