International Journal for the Application of Wireless and Mobile Computing
(IJFAWMC), Volume 4 (October – December, 2018) ISSN: 2141-0720.

THE RISE OF SPAM ACCOUNTS IN MICROBLOGGING SOCIAL NETWORKS - AN EXPERIMENTAL CASE OF THE FEATURES FOR SPAMMER DETECTION

[1]Kayode S. Adewole, [1]Fatima E. Usman-Hamza, [1]Ahmed O. Ameen and [2]Kamaldeen J. Muhammed
[1]Department of Computer Science, University of Ilorin, Nigeria
[2]Department of Educational Technology, University of Ilorin, Nigeria
Email: adewole.ks@unilorin.edu.ng, usman-hamza.fe@unilorin.edu.ng, aminamed@unilorin.edu.ng, jimoh.km@unilorin.edu.ng

Abstract

Microblogging social network, such as Twitter, has become attractive communication media for social spammers to spread malicious contents. As opposed social networks like Facebook and Renren, content distributed on microblogging social networks is unstructured, noisy and short. This characteristic hinders the performance of the traditional semantic analysis technique to effectively detect microblogging spammers. In addition, existing approaches for spammer detection have faced different evasion tactics. In this paper, a framework for identifying spammers on microblogging networks using Twitter as a test bed is proposed. The framework explored a unified feature learning approach by considering five main categories of features. A set of unique features were introduced to complement the existing features in the literature. Eleven (11) supervised machine learning algorithms were trained and tested based on these features. Experimental results demonstrate that Decorate ensemble classifier achieved the best results with an area under the receiver-operating characteristic curve (AUC-ROC) of 0.973 and F-measure of 0.929 using 10-fold cross-validation. Using percentage split, Decorate achieved AUC-ROC of 0.975 and F-measure of 0.940. Experiments were also conducted to investigate the contributions of each category of features. The results indicate that the proposed framework based on the features utilized provides a feasible solution for spammer detection on Twitter microblog.

Keywords**: online social network; microblog; malicious account; spammer; graph mining.**

1.0 INTRODUCTION

Microblogging online social networks, such as Twitter and Tumblr, have become essential part of many people's daily routine. These networks are used for a range of social activities including sharing of interesting contents, locating long-lost friends and interacting within a community that involved families and acquaintances. Microblogging social networks have been in existence for over a decade with huge number of users posting millions of contents on a daily basis (Statista, 2016). Despite the benefits associated with microblogging social networks, these platforms have become attractive communication media for spammers to distribute malicious contents such as spam, phishing links and fake accounts (Chen et al., 2014; Cresci et al., 2017; Lee & Kim, 2014).

Social spam involves the spreading of unsolicited bulk messages for the purpose of promoting advertisements, pornographic websites, fake weight loss, bogus donations, fake news, online job scams and a host of other malicious intents. The rise in spamming activities on Twitter microblog has long been investigated. For instance, Grier et al. (2010) carried out a large-scale analysis of spam, malware, and phishing on Twitter and examined the clickstream activities to external web pages. The authors established that the majority of spam accounts on Twitter are compromised accounts. Thomas et al. (2011) analyzed large collections of suspended accounts from Twitter to identify the behavioral patterns of social spammers. Their findings revealed that about 89% of fraudulent accounts on Twitter rely on unsolicited mentions and trending topics to attract clicks. Cresci et al. (2015) observed that fake accounts have become widely circulated on Twitter to boost account reputations. This creates problems for existing spam account detection solutions to effectively separate spammers from legitimate users (Adewole et al., 2017; Yang et al., 2013).

Several efforts have been committed to reduce the spread of spamming activities on Twitter. For instance, a social network analysis technique was proposed by Ghosh et al. (2012) to distribute trust values as initial seeds based on known spammers and legitimate accounts. The proposed algorithm called Collusion rank assigned two values to the neighbor of the selected accounts which represent trust and untrust values. The problem with this approach is that these values can be easily get diluted, hence, they are inapplicable to categorize accounts as either spam or legitimate. Another line of research is based on machine learning to categorize accounts as spam or legitimate based on certain number of features. For instance, Lee

and Kim (2014) proposed five name-based features from Twitter account group. The problem with this approach focused on the evasion tactics employed by spammers. Thus, it is important to investigate the different features that can be used to identify spammers on microblogging networks.

This paper aims to investigate the performance of five categories of features: contents, user profile, network, timing, and automation in distinguishing spam and legitimate accounts on Twitter microblog. To achieve this objective, a set of unique features were proposed to complement existing features from related studies. Eleven (11) machine learning classifiers were trained and tested using the dataset collected in order to ascertain the performance of the proposed features for detecting spammers on Twitter.

The remainder of this paper is organized as follows: Section 2 discusses related work on spammer detection in social networks. Section 3 focuses on the proposed scheme. Section 4 highlights the results obtained from the various experiments conducted. Section 5 discusses how the proposed model was evaluated. Section 6 concludes the paper and highlights future research areas.

2.0 RELATED WORKS

Several methods have been studied in the literature to detect spam account on social networking sites. These include the use of blacklist, machine learning and graph-based methods (Adewole et al., 2016). For instance, Grier et al. (2010) presented blacklist based method to analyze the links posted on Twitter. The authors stated that 8% of 25 million Uniform Resource Locators (URLs) shared on Twitter are for phishing and malware distribution. Vorakitphan et al. (2018) established that spammers utilized clickbaits strategy to catch the attention of legitimate users on social media platforms. In machine learning domain, Aggarwal et al. (2012) combined content and user profile features to detect phishing URLs on twitter. Martinez-Romo and Araujo (2013) proposed language and content based features to train SVM classifier for malicious tweets detection. Liu et al. (2014) and Zheng et al. (2015) studied spam account detection in Sina Weibo microblogging network. Benevenuto et al. (2010) identified spammers' characteristics related to tweet contents and user-behavior to separate spam from legitimate accounts. Egele et al. (2015) developed COMPA, a system that exploits statistical model and anomaly detection technique to identify compromised accounts on social networks. Lee and Kim (2014) proposed hierarchical clustering approach to initially group spammers with malicious profile names. They trained Markov chain model with valid account names identified from Twitter. Therefore, any account whose characteristic deviates from the identified pattern is flagged as malicious. Chan et al. (2014) proposed new rescaling function to combat adversarial attack on Twitter. Using graph-based method, Ahmed and Abulaish (2012) applied Markov clustering (MCL) algorithm to group a set of profiles as spam and non-spam. The MCL algorithm takes a weighted graph as input and uses random walk approach to assign probabilities to each node on the network which are used to cluster accounts as spam or legitimate. Ghosh et al. (2012) analyzed link farming activities on Twitter and proposed a Collusionrank algorithm to assigned trust level scores to individual account.

As opposed the earlier studies on spam account detection in microblog, this paper investigated five main categories of features including user profile, content, mention network, timing, and automation. Unique set of features were identified to improve the performance of the selected classification algorithms.

3.0 PROPOSED SCHEME

The proposed framework focuses on spam accounts detection on Twitter microblog motivated by the evasion tactics of spammers as previously discussed. The aim is to investigate the performance of the selected classifiers for spam account detection based on the five categories of features identified in this study. To reduce the spread of spamming activities on Twitter, this paper proposes a framework shown in Figure 1.

As an overview of the proposed framework, the process starts by collecting data from Twitter using the Twitter REST API. Content/behavioral features and mentioned graph-based features were extracted to train the selected classification algorithms. Sentiment and term frequency-inverse document frequency (TF-IDF) features were also extracted around the textual contents posted by the users. Specifically, this paper studies the user profile, content, mention network, timing, and automation features to detect spam account on Twitter. The results obtained from the different experiments conducted reveal that the proposed framework is promising for detecting spammers on Twitter microblog. In this study,

Twitter is selected as a test bed for evaluating the proposed framework due to its openness and robust APIs for data collection (Twitter rate limit, 2015).

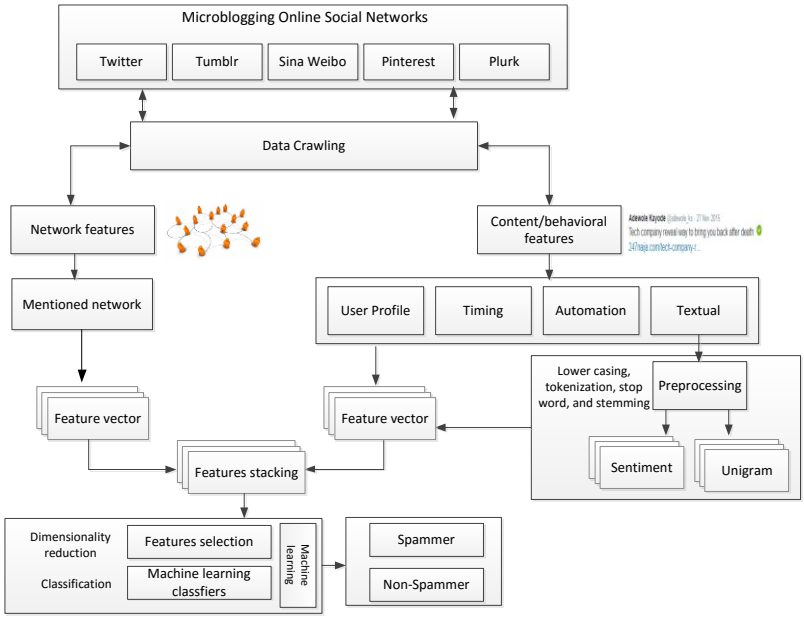


Figure 1: Proposed framework for spammer detection on Twitter microblog.

3.1 Data collection

In order to evaluate the proposed framework for spam account detection on Twitter, a crawler was developed in python using Twitter REST API. The crawling process covers a period of 24 days starting from 20th March to 12th April 2016. The crawlers collected all the tweets posted by the users in the dataset. This study limits the tweets posted by a user to 100 most recent tweets. In total, 3,755,367 tweets were selected with 52,998 distinct accounts as shown in Table 1. It is worth mentioning that, only the tweets authored in English language were selected to give a better understanding of the underlying semantic of the tweets. Table 1 summarizes the dataset used in this study, which shows the total number of tweets, accounts, hashtags, mentions, URLs, mention edges, features, as well as the labeled samples.

Table 1: Summary of the dataset used in this study.

Dataset item	Number of samples
Total Tweets	3,755,367
Total accounts	52,998
Total Hashtag	1,652,405
Total Mention	3,351,656
Total URLs	1,297,288
Mention edges	1,833,086
Total features	69
Spammers identified	3,131
Non-Spammers	4,000
Total labeled samples	7,131

3.2 Features Analyses

This section discusses the five categories of features utilized in this study to distinguish spam accounts from legitimate ones. These features include user profile, content, mention network, timing, and automation.

User profile features: The features captured the basic profile information of an account, such as the number of followers, the number of friends, and so on. The values of these features are extracted from the meta-data returned from Twitter microblog. The user profile features capture the behavioral changes of one account to the other based on their profile contents. Table 2 shows the user profile features used in this study with the additional features introduced to complement the existing ones.

Table 2: Description of user profile features.

Feature name	Description	Reference
Screen name length	The length of the screen name based on characters.	Lee and Kim (2014)
User location	The presence or absence of profile location.	**Proposed**
Profile URL	Whether the user includes URL or not in his profile.	**Proposed**
Age in days	Age of the account in days.	Zheng et al. (2015)
Followers count	Number of followers of the user.	Yang et al. (2013)
Friends count	Number of friends/followees of the user.	Miller et al. (2014)
Statuses count	Total statuses of the account.	**Proposed**
Favourites count	Number of tweets the user has favorited.	Miller et al. (2014)
User description	Indicating presence or absence of profile description.	Aggarwal et al. (2012)
Default profile	When true, indicates that the user has not modified the theme of their profile.	**Proposed**
User Time zone	Indicates presence or absence of time zone.	**Proposed**
Account verified	Indicates whether the account has been verified or not.	Chu et al. (2012)
Default profile image	When true, indicates that the user has not changed the default profile egg avatar.	Alsaleh et al. (2014)
Listed count	The number of the public lists the user is a member.	Miller et al. (2014)
Geo-enabled	Indicates whether or not the user has enabled the possibility of geotagging their tweets.	**Proposed**
Account reputation	Normalized ratio of followers to friends.	Shyni et al. (2016)
Follower following ratio	Ration of the number of follower to friends.	Yang et al. (2013)
Following follower ratio	Ratio of the number of friends to followers.	Zheng et al. (2015)

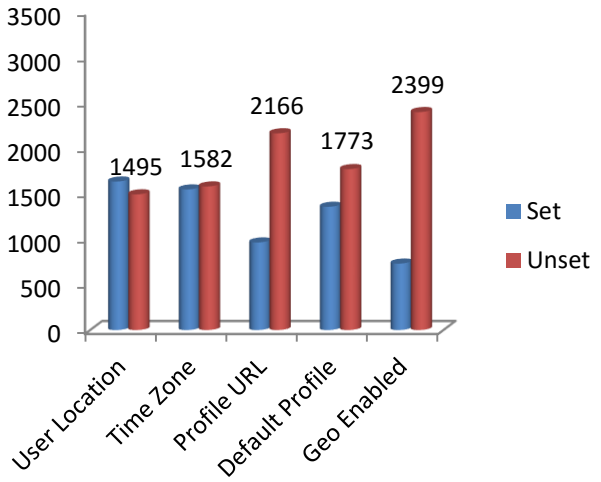


Figure 2: Boolean features for spammers

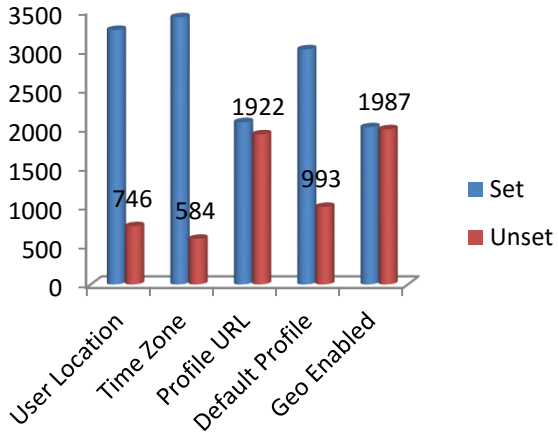
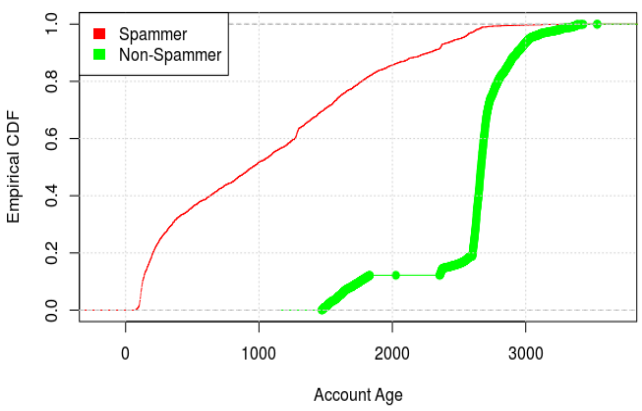
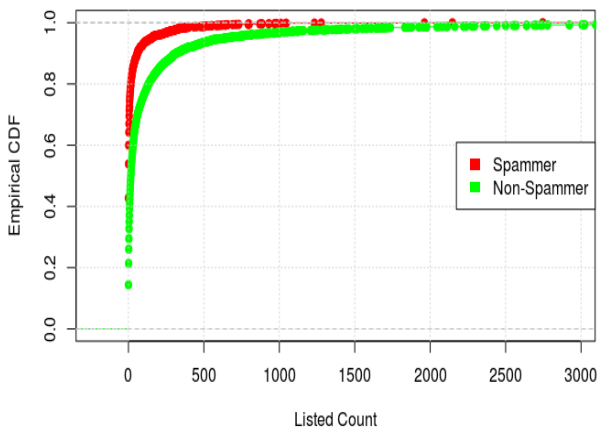


Figure 3: Boolean features for Non-spammers

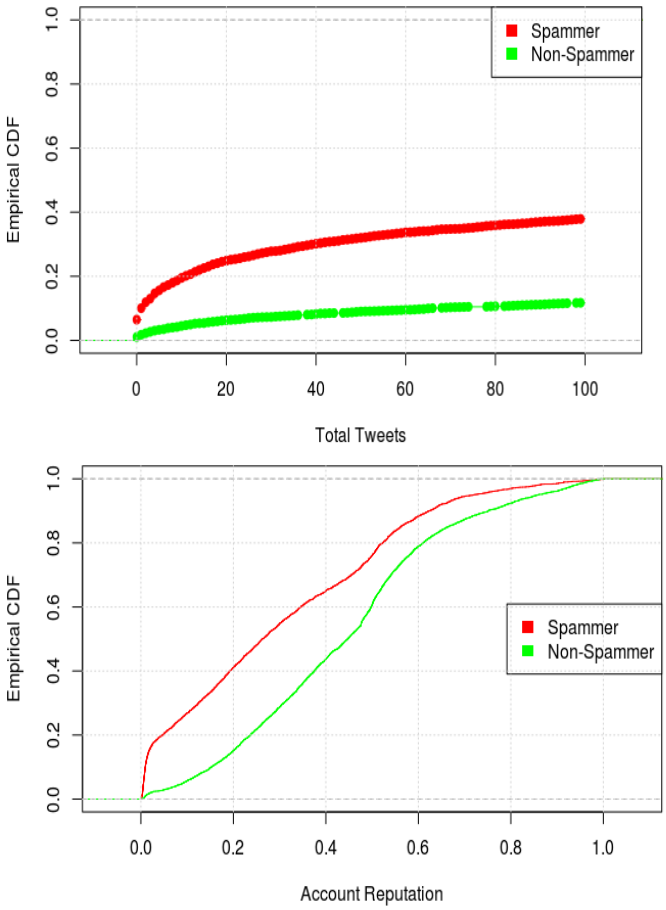
This study investigated the Boolean features in the dataset based on the two classes. It was found that the behavior of spammers and legitimate users differ along these features. The investigation revealed that 47.75%, 50.53%, 69.18%, 56.63%, and 76.62% of spammers do not set their profile location, time zone, URL, theme, and geo-enabled features respectively as shown in Figure 2. Meanwhile, in the case of legitimate users, 18.65%, 14.6%, 48.05%, 24.83%, and 49.68% do not set these features (see Figure 3). The implication of this is that spammers focus more on sending messages along the social structure rather than devoting time to beautify their profiles. In addition, spammers are conscious of disclosing some key information about their accounts in order to avoid detection.



(a)



(b)



(c) (d)

Figure 4: ECDF of: (a) account age (b) Listed count (c) Total tweets (d) Account reputation.

Empirical cumulative distribution (ECDF) plots were constructed to further study the user profile features. Figure 4 shows the ECDF of account age, listed count, total tweets, and account reputation. The findings reveal that spam account exhibits low profile age as compared to legitimate accounts. More than 80% of spammers in the dataset have listed count close to zero. Spam accounts post tweets more than legitimate account, and more than 60% of spammers have account reputation lower than 0.4.

Content-based features: Content-based features study the behavioral patterns of social network accounts around the tweets posted by the users. As evidence in the recent study (Cui, 2016), the use of the traditional bag-of-words models based on the tweet textual content to detect spammers on Twitter has proven inefficient, thus the need to extract content features that can provide discriminative performance is imperative. A set of statistical features were designed as shown in Table 3 to improve the classification results of the selected classifiers.

Table 3: Description of content-based features.

Feature name	Description	Reference
Total tweets	Total tweets sent by the user.	Yang et al. (2013)
Total hashtag	Total number of hashtag used.	Shyni et al. (2016)
Total link	Total number of link posted.	Miller et al. (2014)

Total mention	Total number of users mentioned.	Shyni et al. (2016)
Total retweet	Total number of retweet.	Miller et al. (2014)
Hashtag ratio	Ration of total hashtags to total tweets	Yang et al. (2013)
Link ratio	Ratio of total links to total tweets.	Yang et al. (2013)
Mention ratio	Ratio of total mention to total tweets.	Yang et al. (2013)
Retweet ratio	Ratio of total re-tweet to total tweets.	Yang et al. (2013)
Total tweet favorite count	The number of time the user's tweets has been favorited.	**Proposed**
Deviation of hashtag	Population deviation of hashtags.	**Proposed**
Deviation of link	Population deviation of links.	**Proposed**
Deviation of mention	Population deviation of mentions.	**Proposed**
Deviation of re-tweet	Population deviation of retweets.	**Proposed**
Deviation of tweet length	Population deviation of tweet lengths.	**Proposed**
Deviation of hashtag position aggregate	Population deviation of hashtag position aggregate.	**Proposed**
Deviation of link position aggregate	Population deviation of link position aggregate.	**Proposed**
Deviation of mention position aggregate	Population deviation of mention position aggregate.	**Proposed**
Average daily tweet	Ratio of the total tweet to the number of days between first and last tweets posted.	**Proposed**
Average tweet length	Mean of tweet length.	**Proposed**
Average sentiment polarity	Mean of sentiment polarity for each tweet posted.	**Proposed**
Average sentiment subjectivity	Mean of sentiment subjectivity for each tweet posted.	**Proposed**
Average TF-IDF score	Mean of TF-IDF weight of the tweets.	**Proposed**
Popularity ratio	Ratio of the sum of total tweets favorite and total re-tweet to the number of tweets posted.	**Proposed**
Tweet similarity	Similarity of the tweets text using cosine similarity.	Yang et al. (2013)
Unique URL ratio	Ratio of unique URLs posted to total tweets.	Yang et al. (2013)
Duplicate tweet count	Number of duplicate tweets posted.	Yang et al. (2013)
Unique hashtag	Total number of unique hashtags used.	Shyni et al. (2016)
Unique mention	Total number of unique mentions.	Shyni et al. (2016)
Maximum frequency of hashtag	Maximum value of hashtag frequency.	Shyni et al. (2016)
Average frequency of hashtag	Mean of hashtag used.	Shyni et al. (2016)
Average frequency of mention	Mean of mentions used.	**Proposed**
Average frequency of URLs	Mean of URLs posted.	Shyni et al. (2016)

Network-based features: These features capture the connections or interactions among users on the Twitter microblog. This study modeled users' mentions as a graph *G = (V, E)*, where *V* represents the vertexes and *E* the edges corresponding

to the mention links between users. If a user *u* mentions user *v* in his tweet, we construct an edge *u => v*, which indicates a direct link between *u* and *v*. Thus, the graph *G* is a directed graph that modeled users' mention patterns. A set of graph-based network features were extracted from graph *G* using igraph package in R statistical machine learning tool. In addition, some network features based on the neighborhood were also extracted as defined in the work of (Yang et al., 2013).

Table 4: Description of network features.

Feature name	Description	Reference
Average neighborhood followers	Ratio of sum of the followers of a user's friends to the number of friends of the user.	Yang et al. (2013)
Average neighbor tweets	Ratio of the sum of tweets of a user's friend to the number of friends of a user.	Yang et al. (2013)
Local clustering coefficient of mention	User's local clustering coefficient based on mention network.	**Proposed**
Betweenness centrality of mention	Betweenness centrality of user based on mention network.	**Proposed**
Bidirectional link of mention	Bidirectional link of user based on mention network.	**Proposed**
Bidirectional link ratio of mention	User's bidirectional link ratio from mention network.	**Proposed**
In-degree of mention	User's In-degree from mention graph.	**Proposed**
Out-degree of mention	User's Out-degree from mention graph.	**Proposed**
Degree reputation of mention	Degree reputation based on mention network.	**Proposed**
Degree centrality of mention	Degree centrality of user from mention graph.	**Proposed**
Closeness centrality of mention	User's closeness centrality based on mention network.	**Proposed**
Eigenvector centrality of mention	Eigenvector centrality of user mention network.	**Proposed**
Pagerank of mention	User's Pagerank from mention graph.	**Proposed**

(a) Local clustering coefficient of mention: Local clustering coefficient feature was extracted, which is a useful metric to determine how close a vertex's neighbors are to being a clique. A clique is a small group of people with shared interests. As opposed the work of (Yang et al., 2013), this study focuses on extracting the graph-based features around the mentioned network, which helps in studying the mention relationships among users in the dataset. For each vertex in the mentioned graph *G*, its local clustering score can be computed with Eqn. (1), where K_u is the sum of the in-degree and out-degree of the vertex, and e^u is the total number of edges built by all *u's* neighbors. It was noticed that the local clustering coefficient of spammer based on mentioned network is smaller compared to legitimate users. The reason may be that spammer mentions target users randomly and these accounts may not know each other in reality.

$$LCC(u) = \frac{2|e^u|}{K_u(K_u - 1)} \qquad (1)$$

(b) Betweenness centrality of mention: Betweenness is a centrality measure that uses shortest paths to compute the strength of a vertex in the graph. The metric is obtained using Eqn. (2), where σ_{st} is the total number of shortest paths from node *s* to *t* and $\sigma_{st}(u)$ is the number of those paths that pass through the vertex *u*. *n* is the total number of nodes in graph *G*. Similar to the behavior of spammer as identified in the local clustering coefficient of mentioned network, it was noticed that betweenness centrality of spammer is smaller than the legitimate users.

$$C_B(u) = \sum_{s \neq u \neq t \in V(G)} \frac{\sigma_{st}(u)}{\sigma_{st}} \qquad (2)$$

(c) Bidirectional link of mention: Bidirectional link of mention network defines the total number of links reciprocated by those users mentioned in the tweets. Because spammers randomly mention users in their tweets to launch target attacks, they tend to receive low bidirectional links from the account mentioned as compared to legitimate accounts.

(d) Bidirectional link ratio of mention: Bidirectional link ratio defines the ratio of the number of bidirectional link of a vertex to the total number of out-degree of the vertex. The value is usually low for spammers and high for legitimate users.

(e) In-degree of mention: This feature defines the total number of edges that enters a node. It is computed using Eqn. (3). The value is low for spammers and high for legitimate users.

$$d^{in}{}_u = \sum\nolimits_{[v,u]} G(v,u) \qquad (3)$$

(f) Out-degree of mention: This feature represents the total number of edges that leaves a node. It is computed using Eqn. (4). The value is high for spammers and low for legitimate accounts. The reason is that spammers tend to mention more users for target attacks than legitimate users.

$$d^{out}{}_u = \sum\nolimits_{[u,v]} G(u,v) \qquad (4)$$

(g) Degree reputation of mention: This is the normalized ratio of the In-degree to the Out-degree of a vertex. The value of degree reputation of mention for spammers is low compared to the degree reputation of legitimate users. The feature is computed as shown in Eqn. (5).

$$dr(u) = \frac{\left| d^{in}{}_u \cup d^{out}{}_u \right|}{\left| d^{in}{}_u \right|} \qquad (5)$$

(h) Degree centrality of mention: This feature defines the sum of the total In-degree and Out-degree of a vertex. The degree centrality of spammers based on the mention network is low compared to legitimate accounts as observed during analysis. Eqn. (6) shows how to compute degree centrality for a vertex.

$$\deg(u) = d^{in}{}_u \cup d^{out}{}_u \qquad (6)$$

(i) Closeness centrality of mention: The closeness centrality metric measures the importance of a vertex based on how close a given vertex is to the other vertices in the graph. The most center vertices are important as they can reach the whole network more quickly than non-central vertices. This can be utilized to measure the quality of the connection of a node within the network. Closeness centrality metric can be obtained using Eqn. (7), where *d(v,u)* is the distance between vertices *v* and *u*. It was noticed that the closeness centrality of spammers based on the mention network is low compared to legitimate accounts.

$$C(u) = \frac{1}{n-1} \sum\nolimits_{v \in V(G)} d(v,u) \qquad (7)$$

(j) Eigenvector centrality of mention: This is useful for measuring how the centrality of a node depends on its neighbors' centralities. The metric does not only measure how the vertex is position within the network, but also the quality of the links built with the vertex neighbors. Eqn. (8) shows how the eigenvector centrality is computed from the mentioned graph, where $EC(v_j)$ is the eigenvector of the vertex v_j connected to *u*, $A=[a_{ij}]$ is the adjacency matrix, and λ is a constant. The EC of one vertex relies on the *EC* of another vertex it is connected to. The *EC(v)* is calculated by finding the eigenvector

associated with the highest eigenvalue according to Perron-Frobenious theorem (Ferrara & Fiumara, 2012). The ith entry of the vector corresponds to the eigenvector centrality score of ith vertex. The value of eigenvector of spammers is low compared to legitimate users.

$$EC(u_i) = \frac{1}{\lambda}\sum_{j=1}^{N} a_{ij} EC(v_j) \qquad (8)$$

(k) PageRank of mention: The Google PageRank is a modified version of eigenvector centrality metric. PageRank of a vertex *u* relates the PageRank of the vertex it is connected to in the graph. Eqn. (9) shows how PageRank score is obtained from the graph, where $d = 0.85$ is the damping factor. *N* is the total number of vertices considered in the mentioned graph, $PR(v_j)$ is the PageRank of the vertex v_j, $M(u)$ is the set of vertices that link to vertex *u*. $L(v_j)$ is the number of outbound links of vertex v_j. This study found that spammers have low PageRank score compared to legitimate accounts.

$$PR(u) = \frac{1-d}{N} + d\sum_{v_j \in M(u)} \frac{PR(v_j)}{L(v_j)} \qquad (9)$$

Timing-based features: Timing-based features deal with the tweeting rate and following rate of an account. The features examine the posting and following patterns of users on the Twitter microblog. This study adopted the timing-based features from the related studies. Table 5 shows the description of the two features in this category. Spammer follows a large number of users and generates more tweets than the legitimate users.

Table 5: Description of timing-based features.

Feature name	Description	Reference
Following rate	Ratio of the number of friends to the age of an account.	Yang et al. (2013)
Tweeting rate	Ratio of the total number of tweets to the age of the account.	Yang et al. (2013)

Automation-based features: Similar to the timing-based features, this study adopted automation features utilized in (Yang et al., 2013). Yang et al. (2013) established that spammers utilize automation technique for posting tweets due to the high cost of manually maintaining many spam accounts. Table 6 shows the description of automation-based features adopted in this study.

Table 6: Description of automation-based features.

Feature name	Description	Reference
API ratio	Ratio of the number of tweets sent using API to total number of tweets.	Yang et al. (2013)
API URL ratio	Ratio of the number of tweets sent using API that contains URL to the total number of tweets sent using API.	Yang et al. (2013)
API tweet similarity	Number of similar tweets sent using API.	Yang et al. (2013)

4.0 EXPERIMENTAL SETUP

This study employed eleven (11) classification algorithms to evaluate the effectiveness of the proposed features. These algorithms include Random Forest, J48, ADTree, SVM (Sequential minimal optimization), Multilayer perceptron (MLP), AdaBoost, Decorate, LogitBoost, Bayes Network, Voting feature interval (VFI), and Random committee. All experiments were conducted using WEKA, a machine learning tool developed at the University of Waikato in New Zealand (WEKA, 2016). The implementation of the classification algorithms on WEKA was done with the heap memory set to 2GB on Ubuntu 14.04

operating system. The system has a random access memory (RAM) of 20GB and 3.40GHz Intel Core i7 CPU. Eleven (11) classification algorithms were used with the aim of finding the best classifiers as well as the feature subsets that will produce improved classification results. Standard evaluation metrics such as detection rate, false positive rate, precision, recall, F-measure and ROC were employed to evaluate the results obtained from each experiment.

4.1 Results and discussions

As earlier discussed, this study examines the performance of eleven (11) classification algorithms based on the features utilized in this paper. The subsequent sections present the results obtained for each experiment conducted.

4.1.1 Classification performance with all features

This study first conducted an experiment using all the features discussed in the previous sections. The result of this experiment shows that Decorate, an ensemble-based classification algorithm, outperformed other classifiers. Two training methods were used during the experiments based on 10-fold cross-validation and percentage split in order to assess the behaviors of the selected classifiers across different training methods. In 10-fold cross-validation, the labeled samples are divided into 10 subsets of equal size. In each round of the training, one out of 10 subsets is held as the testing set to validate the classifier, while the remaining nine subsets are used to train the classification algorithm. In percentage split, the labeled are partitioned into two sizes using percentage as the basis for partitioning. 80% of the samples was utilized for training and 20% for validating the classifiers. The result of the percentage split performs better than the 10-fold cross-validation in this experiment. As shown in Table 7, the results of the two methods achieved promising performance based on the detection rate (DR), false positive rate (FPR), precision, recall, F-measure, and AUC-ROC values. This study achieved AUC-ROC of 97.50% with percentage split and 97.30% using 10-fold cross-validation method. This result indicates that the classification models provide a reasonable solution to detect spammers on Twitter.

Table 7: Classification results with all the features.

Classifiers	10-fold cross-validation						Percentage split (80% train and 20% test)					
	DR	FPR	Precision	Recall	F1	ROC	DR	FPR	Precision	Recall	F1	ROC
Random Forest	0.926	0.084	0.927	0.926	0.925	0.973	0.935	0.073	0.936	0.935	0.935	0.976
J48	0.928	0.083	0.929	0.928	0.927	0.963	0.935	0.077	0.938	0.935	0.934	0.973
ADTree	0.928	0.087	0.932	0.928	0.927	0.973	0.935	0.074	0.939	0.935	0.934	0.973
SVM (SMO)	0.874	0.128	0.874	0.874	0.874	0.874	0.882	0.118	0.883	0.882	0.882	0.882
MLP	0.916	0.097	0.919	0.916	0.916	0.968	0.922	0.088	0.924	0.922	0.922	0.970
AdaBoost	0.926	0.084	0.927	0.926	0.926	0.974	0.934	0.074	0.935	0.934	0.934	0.978
Decorate	**0.929**	**0.083**	**0.931**	**0.929**	**0.929**	**0.973**	**0.940**	**0.070**	**0.943**	**0.940**	**0.940**	**0.975**
LogitBoost	0.927	0.091	0.933	0.927	0.926	0.971	0.931	0.083	0.935	0.931	0.930	0.973
BayesNet	0.880	0.120	0.881	0.880	0.881	0.942	0.886	0.112	0.888	0.886	0.887	0.940
VFI	0.866	0.171	0.890	0.866	0.861	0.895	0.873	0.158	0.896	0.873	0.869	0.906
Random committee	0.921	0.089	0.922	0.921	0.921	0.967	0.929	0.080	0.930	0.929	0.929	0.967

As shown in Table *7*, it was observed that tree-based and ensemble-based classification algorithms achieved better performance and can be used to separate spammers from legitimate users on the Twitter microblog. The least performed classifier in terms of AUC-ROC is SVM (SMO) algorithm, which achieves AUC-ROC of 87.40% using 10-fold cross-validation and 88.20% using percentage split method.

4.1.2 Classification performance based on feature categories

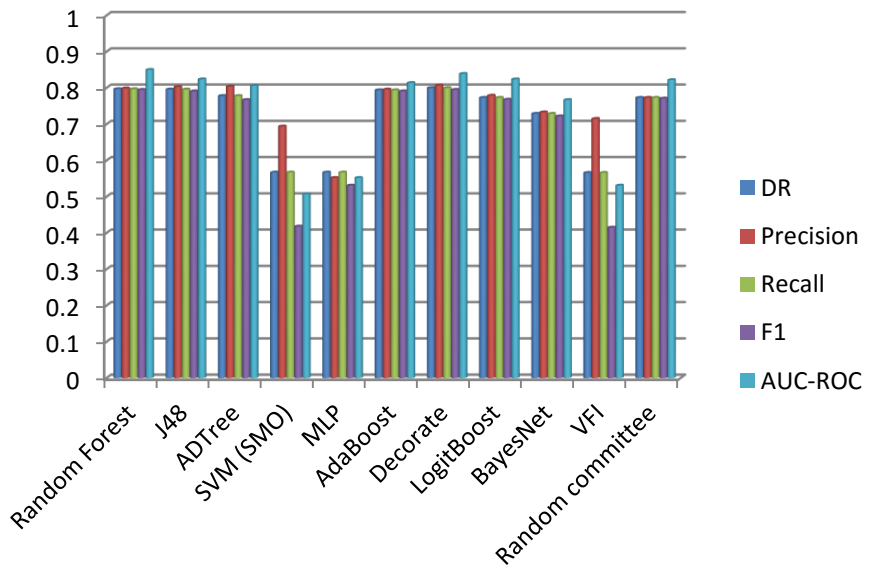
This section presents the results of experiments to rank each feature category in isolation. Each category was trained and validated based on the selected classifiers. Summarily, it was noticed that the huge contribution to the classification performance was obtained from the user profile based features with LogitBoost achieving AUC-ROC of 97.0% and F-measure of 92.7% using 10-fold cross-validation (see Table *8*). The implication of this result is that the behavioral profile of

accounts taking into consideration the features identified in this study provides a clear separation between spammers and legitimate accounts.

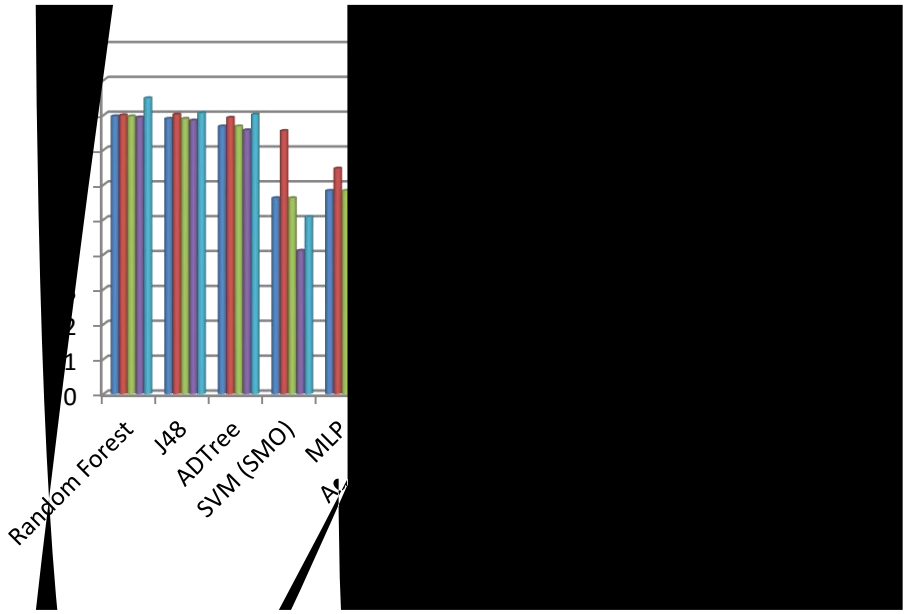
Table 8: Classification based on user profile features.

Classifiers	10-fold cross-validation						Percentage split (80% train and 20% test)					
	DR	FPR	Precision	Recall	F1	ROC	DR	FPR	Precision	Recall	F1	ROC
Random Forest	0.924	0.086	0.925	0.924	0.923	0.973	0.927	0.080	0.928	0.927	0.927	0.977
J48	0.924	0.086	0.924	0.924	0.923	0.943	0.919	0.086	0.919	0.919	0.919	0.939
ADTree	0.926	0.091	0.931	0.926	0.925	0.972	0.932	0.083	0.938	0.932	0.931	0.973
SVM (SMO)	0.878	0.122	0.879	0.878	0.878	0.878	0.888	0.111	0.889	0.888	0.888	0.888
MLP	0.900	0.112	0.901	0.900	0.899	0.965	0.903	0.100	0.902	0.903	0.903	0.966
AdaBoost	0.923	0.087	0.924	0.923	0.923	0.973	0.926	0.080	0.927	0.926	0.926	0.962
Decorate	0.920	0.090	0.921	0.920	0.920	0.972	0.927	0.080	0.928	0.927	0.927	0.973
LogitBoost	**0.928**	**0.091**	**0.935**	**0.928**	**0.927**	**0.970**	**0.931**	**0.083**	**0.935**	**0.931**	**0.930**	**0.973**
BayesNet	0.865	0.135	0.866	0.865	0.865	0.931	0.874	0.124	0.875	0.874	0.874	0.932
VFI	0.860	0.161	0.865	0.860	0.858	0.897	0.868	0.147	0.871	0.868	0.867	0.910
Random committee	0.919	0.091	0.920	0.919	0.919	0.967	0.919	0.089	0.920	0.919	0.919	0.971

The performance of automation-based features follows the user profile features with the results indicating that automation behavior of spammer is different from legitimate users. The result of this classification shows that Decorate produces the best result based on 10-fold cross-validation method. As shown in Figure 5(a), Decorate achieves AUC-ROC of 83.9% and F-measure of 79.5% using automation features. This study observed very close performance with Random Forest and AdaBoost classifiers based on percentage split (see Figure 5(b)).



(a) 10-fold cross-validation

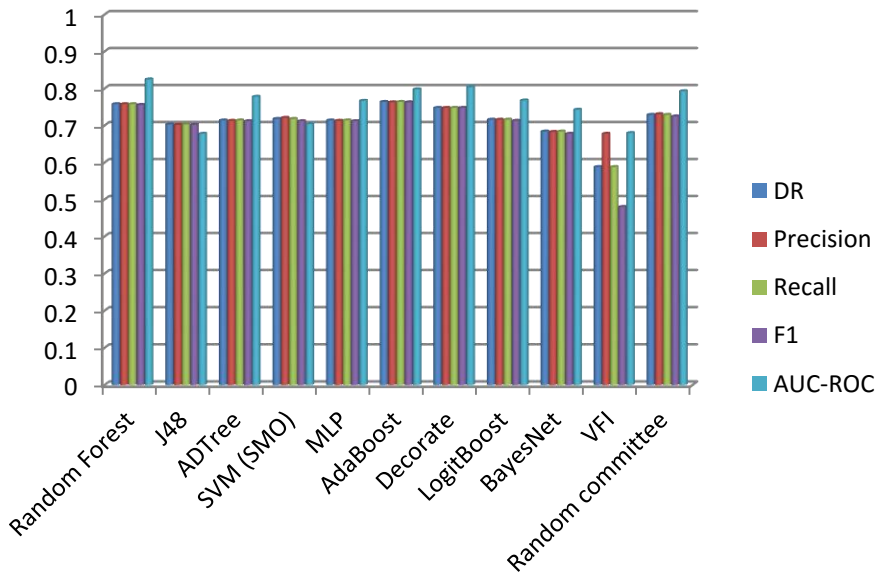


(b) Percentage split

Figure 5: Classification result based on automation features (a) 10-fold cross-validation (b) Percentage split.

The third perform feature category is content-based features. In this case, this study obtained the best result with Random Forest classifier. Random Forest achieved AUC-ROC of 82.5% and F-measure of 75.9% using the content-based features in 10-fold cross-validation. The performance of AdaBoost classifier is very close to the Random Forest. Figure 6(a) summarizes the results. In this experiment, the least performed classifier is VFI in both 10-fold cross-validation and percentage split methods (see Figure 6(b)).

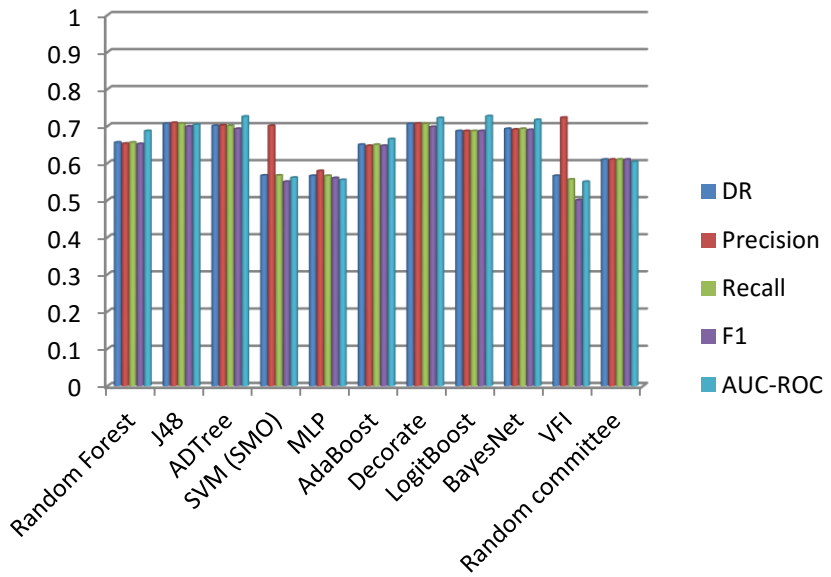
(a) 10-fold cross-validation



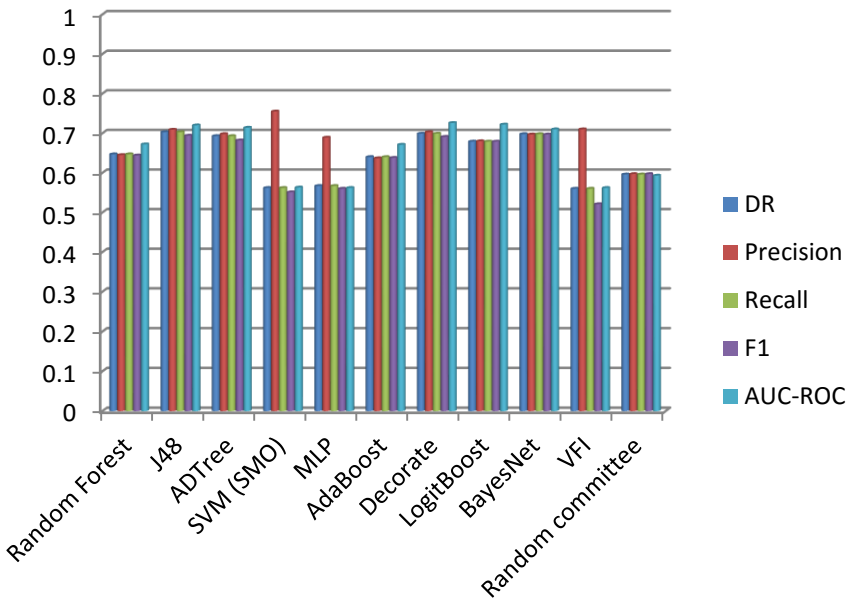
(b) Percentage split

Figure 6: Classification based on content features (a) 10-fold cross-validation (b) Percentage split.

The forth-ranked feature category based on classification results is timing-based features with J48 achieving the best result. J48 algorithm produces AUC-ROC of 70.4% and F-measure of 69.9% using 10-fold cross-validation as shown in Figure 7(a). The implication of this result is that spammers' following and tweeting activities based on their respective profile age deviates from the normal accounts behaviors. In this experiment, VFI classifier achieves the least classification accuracy.



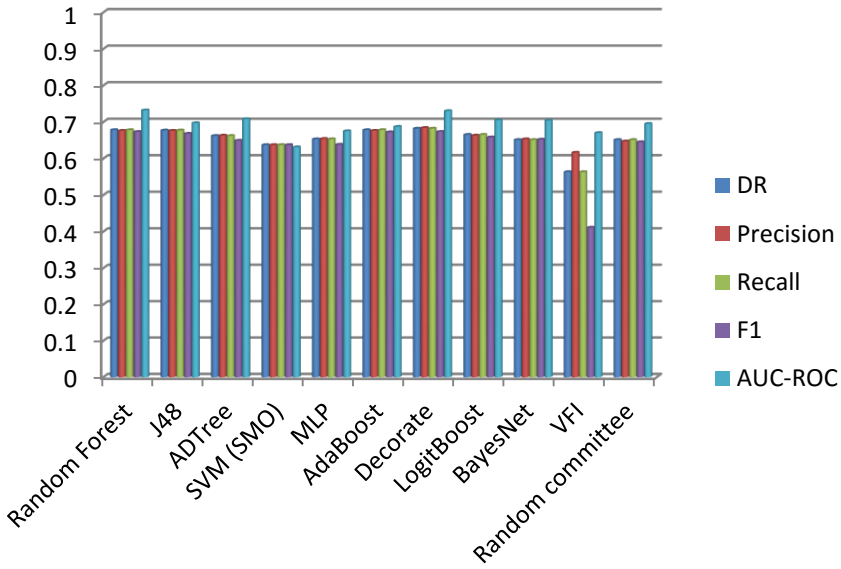
(a) 10-fold cross-validation



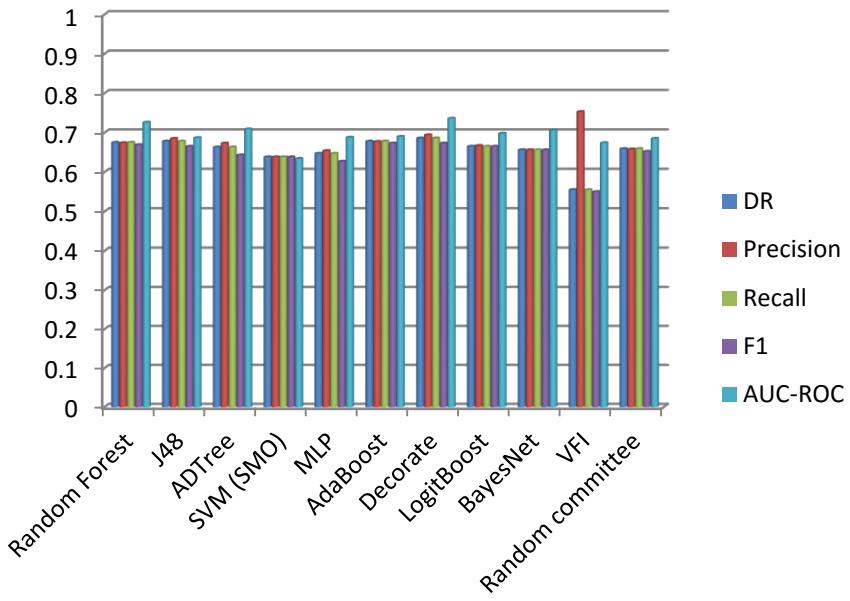
(b) Percentage split

Figure 7: Classification result using timing-based features (a) 10-fold cross-validation (b) Percentage split.

The fifth-ranked feature category based on classification results is the mentioned graph-based network features. In this experiment, best result was obtained with Decorate classifier, which produces an AUC-ROC of 73.0% and F-measure of 67.3% using 10-fold cross-validation as summarized in Figure 8(a). This result indicate that the mention behavior of spammer also deviate from the legitimate users. The least performed classifier in terms of F-measure and recall is VFI.



(a) 10-fold cross-validation



(b) Percentage split

Figure 8: Classification result using mentioned graph-based network features (a) 10-fold cross-validation (b) Percentage split.

5.0 MODEL EVALUATION

This section provides information on the method employed to evaluate the proposed classification model. Since Decorate classifier produced the best performance results based on the overall features utilized in this study and the complete labeled samples, a confusion matrix was constructed shown in Table 9 to evaluate the classification model in order to ascertain how the model is able to distinguish spammers from non-spammers. This matrix indicates that the Decorate classifier is able to detect 2,731 accounts as spammers out of 3,131 total spammers' accounts, achieving DR of 87.2% using 10-fold cross-validation. In the case of non-spammers, Decorate detects 3,894 accounts as non-spammers out of the total accounts of 4,000. This result produces a DR of 97.4% with 10-fold cross-validation. In the case of percentage split method using training and testing samples of 80% and 20% respectively, Decorate identified 565 spammers out of the total of 653 spammers accounts used for testing. This classification result produces a DR of 89.0%. For non-spammers accounts, Decorate identified 776 accounts as non-spammers out of the total of 791 accounts producing a DR of 98.1% for non-spammer detection. These results confirmed that the proposed solution is visible for spammer detection in Twitter microblog.

Table 9: Confusion matrix for Decorate classifier using 10-fold cross-validation and percentage split with all the features.

	Predicted (10-fold cross-validation)		Predicted (percentage split 20% test)	
Actual	Spammer	Non-spammer	Spammer	Non-spammer
Spammer	2,731	400	565	70
Non-spammer	106	3,894	15	776

6.0 CONCLUSION

Spam accounts detection problem in online social networks has become a continuous battle between spammers and detection models. The significant increase in evasion tactics with existing filters has called for the need to investigate the new approach for detecting social spammers. This paper proposed a framework for spam account detection using Twitter microblog as a test bed. This research extensively studied five categories of features for distinguishing spammers from

legitimate users. In addition to the use of content/behavioral based features, this study investigated the mentioned network of spammers by extracting different graph-based features around the mentioned graph constructed. It was noticed that the mention behavior of spammers based on these features differs from the legitimate users. Eleven (11) classification algorithms were evaluated to identify the classification model that can produce the best results for the problem under investigation. The results show that ensemble and tree based classifiers produced good models for spam accounts detection task. More specifically, Decorate ensemble classifier using 10-fold cross-validation and percentage split produced the best accuracy. By considering all the features identified in this study, Decorate classifier produced AUC-ROC of 97.3% and 97.5% using 10-fold and percentage split respectively. This research further investigated the performance of the classification models based on the category of features proposed in this study. The results of the experiments revealed that user profile features contributed immensely to the overall classification performance. This is followed by automation, contents, timing, and mention graph-based features. One of the future directions is to further investigate the combination of the features that can better improve the classification results using meta-heuristic optimization approach and more labeled samples.

REFERENCES

Adewole, K. S., Anuar, N. B., Kamsin, A., & Sangaiah, A. K. (2017). SMSAD: a framework for spam message and spam account detection. *Multimedia Tools and Applications*, 1-36.

Adewole, K. S., Anuar, N. B., Kamsin, A., Varathan, K. D., & Razak, S. A. (2016). Malicious accounts: Dark of the social networks. *Journal of Network and Computer Applications*. doi: http://dx.doi.org/10.1016/j.jnca.2016.11.030

Aggarwal, A., Rajadesingan, A., & Kumaraguru, P. (2012). *PhishAri: Automatic realtime phishing detection on twitter.* Paper presented at the eCrime Researchers Summit (eCrime), 2012.

Ahmed, F., & Abulaish, M. (2012). *An MCL-based approach for spam profile detection in online social networks.* Paper presented at the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom).

Alsaleh, M., Alarifi, A., Al-Salman, A. M., Alfayez, M., & Almuhaysin, A. (2014). *TSD: Detecting Sybil Accounts in Twitter.* Paper presented at the 2014 13th IEEE International Conference on Machine Learning and Applications (ICMLA).

Benevenuto, F., Magno, G., Rodrigues, T., & Almeida, V. (2010). *Detecting spammers on Twitter.* Paper presented at the 7th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference, CEAS 2010.

Chan, P. P. K., Yang, C., Yeung, D. S., & Ng, W. W. Y. (2014). Spam filtering for short messages in adversarial environment. *Neurocomputing, 155*, 167-176. doi: 10.1016/j.neucom.2014.12.034

Chen, C.-M., Guan, D., & Su, Q.-K. (2014). Feature set identification for detecting suspicious URLs using Bayesian classification in social networks. *Information Sciences, 289*, 133-147.

Chu, Z., Gianvecchio, S., Wang, H., & Jajodia, S. (2012). Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing, 9*(6), 811-824. doi:10.1109/TDSC.2012.75

Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., & Tesconi, M. (2015). Fame for sale: Efficient detection of fake Twitter followers. *Decision Support Systems, 80*, 56-71. doi: 10.1016/j.dss.2015.09.003

Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., & Tesconi, M. (2017). The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. *arXiv preprint arXiv:1701.03017*.

Cui, X. (2016). Identifying Suspended Accounts In Twitter. http://scholar.uwindsor.ca/etd/5725/

Egele, M., Stringhini, G., Kruegel, C., & Vigna, G. (2015). Towards Detecting Compromised Accounts on Social Networks. *IEEE Transaction on Dependable and Secure Computing*. doi: 10.1109/TDSC.2015.2479616

Ferrara, E., & Fiumara, G. (2012). *Mining and Analysis of Online Social Networks.* (PhD doctoral thesis), University of Messina, Italy.

Ghosh, S., Viswanath, B., Kooti, F., Sharma, N. K., Korlam, G., Benevenuto, F., . . . Gummadi, K. P. (2012). Understanding and combating link farming in the twitter social network. *Proceedings of the 21st International Conference World Wide Web*, 61.

Grier, C., Thomas, K., Paxson, V., & Zhang, M. (2010). @spam: the underground on 140 characters or less. *Proceedings of the 17th ACM conference on Computer and communications security*, 27-37.

Lee, S., & Kim, J. (2014). Early filtering of ephemeral malicious accounts on Twitter. *Computer Communications, 54*, 48-57.

Liu, Y., Wu, B., Wang, B., & Li, G. (2014). SDHM: A Hybrid Model for Spammer Detection in Weibo. *2014 Proceedings of the Ieee/Acm International Conference on Advances in Social Networks Analysis and Mining (Asonam 2014)*, 942-947.

Martinez-Romo, J., & Araujo, L. (2013). Detecting malicious tweets in trending topics using a statistical analysis of language. *Expert Systems with Applications, 40*, 2992-3000. doi: 10.1016/j.eswa.2012.12.015

Miller, Z., Dickinson, B., Deitrick, W., Hu, W., & Wang, A. H. (2014). Twitter spammer detection using data stream clustering. *Information Sciences, 260*, 64-73. doi: 10.1016/j.ins.2013.11.016

Shyni, C. E., Sundar, A. D., & Ebby, G. S. E. (2016). Spam profile detection in online social network using statistical approach. *Asian Journal of Information Technology, 15*(7), 1253-1262.

Statista. (2016). Leading social networks worldwide as of April 2016, ranked by number of active users (in millions). from http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

Thomas, K., Grier, C., Song, D., & Paxson, V. (2011). *Suspended accounts in retrospect: an analysis of twitter spam.* Paper presented at the Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference.

Twitter rate limit. (2015). Twitter rate limit for search/tweets REST API calls. from https://dev.twitter.com/rest/public/rate-limits

Vorakitphan, V., Leu, F.-Y., & Fan, Y.-C. (2018). *Clickbait Detection Based on Word Embedding Models.* Paper presented at the International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing.

WEKA. (2016). The University of Waikato. Retrieved 2nd February, 2016, from http://www.cs.waikato.ac.nz/ml/weka/

Yang, C., Harkreader, R., & Gu, G. (2013). Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers. *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, 8*(8), 1280-1293. doi: 10.1109/tifs.2013.2267732

Zheng, X., Zeng, Z., Chen, Z., Yu, Y., & Rong, C. (2015). Detecting spammers on social networks. *Neurocomputing, 159*, 27-34. doi: 10.1016/j.neucom.2015.02.047