Detecting Spam in Twitter Microblogging Services: A Novel Machine Learning Approach based on Domain Popularity

Khalid Binsaeed[1], Gianluca Stringhini[2]
[1]College of Computer and Information Sciences
King Saud University, Riyadh, KSA
[2]Dept. of Computer Science
University College London, London, UK

Ahmed E. Youssef[3]
College of Computer and Information Sciences
King Saud University, Riyadh, KSA
Dept. of Computers and Systems Engineering, Faculty of
Engineering at Helwan, Helwan University, Cairo, Egypt

***Abstract*—Detecting Internet malicious activities has been and continues to be a critical issue that needs to be addressed effectively. This is essential to protect our personal information, computing resources, and financial capitals from unsolicited actions, such as, credential information theft, downloading and installing malware, extortion, etc. The introduction of the social media such as Twitter has given malicious users a new and a promising platform to perform their activities, ranging from a simple spam message to taking a full control over the victim's machine. Twitter revealed that its algorithms for detecting spam are not very effective; most of the trending hashtags include unrelated spam and advertising tweets which indicates that there is a problem with the currently used spam detection framework. This paper proposes a new approach for detecting spam in Twitter microblogging using Machine Learning (ML) techniques and domain popularity services. The proposed approach comprises two main stages: 1) Tweets are collected periodically and filtered by selecting the ones that appear more frequently than a decided threshold in the specified period (i.e. common tweets). Then, an inspection is conducted on the common tweets by checking the associated URL domain with Alexa's top one million globally viewed websites. If a tweet is common on Twitter but does not appear on the top one million globally viewed websites, it is flagged as a potential spam. 2) The second stage kicks in by running ML algorithms on the flagged tweets to extract features that help detect the cluster of spam and prevent it in real-time. The performance of the proposed approach has been evaluated using three most popular classification models (random forest, J48, and Naïve Bayes). For all classifiers, results showed the effectiveness of the proposed method in terms of different performance metrics (e.g. precision, sensitivity, F1-score, accuracy) and using different test scenarios.**

Keywords*—Spam detection; phishing detection; domain popularity; machine learning; Twitter

I. Introduction

Nowadays, the relationship between people and the Internet has changed dramatically; social media and microblogging services have taken an essential part in the way we live. From a statistical point of view Alexa's website of the global top 500 most visited sites has shown that five websites of the top twenty-five websites are related to social media [1]. This fact supports the claim that social media sites are amongst the most visited around the world. The wide spread of social media has attracted spammers and hackers to perform their activities on these platforms, giving them a huge opportunity and an easy way to reach networks of users who are potentially good targets; and due to the openness of the design of social media, users trust each other on their networks even if they are controlled by hackers. Although social media have given spam and phishing the ideal environment to live in, malicious activities were popular before that; their main target back was electronic mails and web services such as forums. However, the peak point was not reached until social media sites were introduced. In [2], the authors gave a reason for that, they mentioned that the built by design trust relationship between users of these services gave more confidence to the user to read and/or click on hyperlinks sent by a friend on that service. This fact is appealing for the attacker as if he controls one victim, his friend list will be likely trusting his messages.

In [3] the author reported that the spam on social media sites has raised by 355% in the first half of 2013, he justified that as "spammers are turning to the fastest growing communication media to circumvent traditional security infrastructures that were used to detect email spam". He also reported the impact of spam as "the impact of social media spam is already significant, it can damage brand appearance and turns fans and followers into foes". These facts motivate the necessity for developing effective algorithms to prevent spam and phishing on microblogging services, and in order to do that, an effective detection method must be placed first, then the prevention could be done. Almost all techniques in the industry relies on detecting before preventing. Section 3 in this paper describes the current methods used for spam detection in detail.

According to [1], Twitter website is now the most popular microblogging service on the Internet. In contrast to other social media services on the Internet, Twitter has shown, since its introduction in 2006, that it can be an appealing service to almost every user of the Internet; it can be a foundation for blogging, socializing, news, political activities, knowledge and/or job hunting. The feature that makes Twitter distinct from other social media sites that provide the same services is the privacy by design. This feature allows users to get all services without being obligated to reveal any information about themselves or having any user following them. This is given by the nature of the relation between users (unidirectional) that allows the user to follow any other user without being forced to let them follow you back. This nature is interesting to malicious users since it will allow them to spread their malicious content on the network without having to friend a single user; meaning that the other users on the network will still see their tweets without the need to have the attacker follow the victim; for example, by searching of hashtags.

Another important feature of Twitter is the hashtags; a hashtag implies grouping similar tweets together in a way that allows users to browse them based on a specific subject. This will help attackers to get the highest views possible by targeting popular subjects (e.g. sports, politics, gaming, etc.) and tweeting their messages into them. Hashtags will aid in reaching potentially all users of the service, each according to his/her own interest. This is a crucial evolution in the way spam is spreading; the attacker does not even need to know the target address or name.

Due to the aforementioned reasons, Twitter's algorithms for detecting spam are not very effective since most of the trending hashtags include unrelated spam and advertising tweets. This indicates that there is a problem with the currently used spam detection framework. Hence, many researchers are concerned with investigating and solving the problem of detecting/preventing spam and phishing on Twitter platform [16,24,29-40]. This paper introduces a new approach for detecting spam on microblogging services using domain popularity and Machine Learning (ML) algorithms. The proposed approach comprises two stages: 1) tweets are collected periodically and filtered by selecting the tweets that appear with a frequency more than a decided threshold in a specified period; these tweets are called common tweets. After that, an examination is conducted on the common tweets by checking their associated URL domain with Alexa's top one million globally viewed websites. If a tweet is common on Twitter but does not appear on the top one million globally viewed websites (e.g. google.com), it is flagged as a potential spam. 2) The second stage kicks in by running ML algorithms on the flagged messages to extract features that can help detect the cluster of spam and prevent it in real-time. The performance of the proposed approach has been evaluated through extensive experiments using three different classification models (random forest, J48, and Naïve Bayes). For all classifiers, results showed the effectiveness of the proposed method in terms of different performance metrics (e.g. precision, sensitivity, F1-score, accuracy) and using different test scenarios.

The rest of this paper is organized as follows: Section 2 gives an essential background on spam detection and Section 3 reviews the related work. In Section 4, we present the proposed approach in detail. In Section 5, the performance of the proposed approach is evaluated through extensive experiments. In Section 6, we discuss operation and limitations of the proposed approach. Finally, in Section 7, we conclude this work and give future research perspectives.

II. Background

Spam and phishing are now spreading faster than ever which means that all users on the Internet are potential targets. This is true since spam and phishing messages are designed to exploit the trust concept of the system; meaning that they will use genuine techniques (e.g. sending email) to spread across networks. In this section, we give essential background relevant to spam and phishing.

A. Spam

Oxford dictionary [4] defines spam as "Irrelevant or inappropriate messages sent on the Internet to a large number of recipients". From this description, we can realize that the messages are sent in the network to a group of recipients, this means that a single user receiving a spam message is most likely not interested in the message. The objective of spam varies depending on the intention of the spammer. Some spammers intend to spread malware; others use spams to build a botnet; or for other objectives based on the interest of the spammer. However, the largest use for spam is in the advertisement industry. In [5], the authors reported "...we estimate that spammers and spam-advertised merchants collect gross worldwide revenues on the order of $200 million per year…", they proceeded by showing why e-spam advertisement can be profitable, they argued that unlike post mail spam, the cost associated with using technology to spread spam is negligible. Still, this does not excuse the depraved side of spam. Spam still leads to wasting the victim's time or losing productivity of a service (e.g. Twitter hashtags).

The idea of spam is not exclusive to the Internet; spam was used before the creation of the Internet. The network back then was between universities and large government sites and spam was used on those networks. Nonetheless, it was easy to contain and was not problematic at that time. During the 90s the age of the Internet began, it was commercialized and used by the public within their home. In [6], the authors reported, "By the spring of 1996 spam made up a significant portion of the email received by customers of the major Internet service providers…"; since that date spam was recognized by the industry as a problem that need to be solved. Researchers began to develop new ways to deal with spam; for example, Microsoft began developing research to filter spam via machine learning, they found that the spam messages share some similar characteristic and it is possible to detect a spam message from a legitimate message, they were able to eliminate a large portion of junk mail just by observing the mail stream [7]. Although it was not a solution, at the time, this was an achievement.

The ease of performing spam on online social media helped increase its appearances in this platform. Still, this is only one of the many possible reasons for the popularity of spam in social media. In [8] the authors mentioned that spam on social media is highly effective and this attracts spammers. On the other hand, in [2], the authors believe that the abuse of trust between users of the services is the cause for spam. Furthermore, the authors of [9] found after analyzing a group of spam accounts on Twitter that more than half of them were genuine accounts at some point in time and then they were compromised by the spammer. This last finding can be used to support the one before it (exploitation of trust) since the compromised user accounts will exploit the trust of his/her friends. From the aforementioned discussion, it is not hard to see how and why social media are perfect platform for spam, they are faster, more scalable, and more effective than traditional spam. All of the previous findings are just some of many possible reasons to why spam is popular on social media as opposed to other traditional ways.

B. Phising

Phishing is a part of the social engineering cluster of attacks where the attacker tries to trick the victim into stealing their sensitive information by sending a message pretending to be a legitimate entity. There is a variety of phishing techniques that can be done through email, SMS, or using fake websites. Phishing can also come in a verity of types, for instance, if the attack is directed to a specific person it is called spear phishing. Nonetheless, for the purpose of this work the term phishing will always denote the general type of phishing.

In [10], the authors reported that 5% of the attackers are successful in convincing their victims. Two years later another group of researchers conducted an in-depth study on phishing [11], they used 20 websites and brought 22 participants, they started asking the participants which of the 20 websites is fake. As expected, 90% of the participants failed to identify the phishing websites from the legitimate ones. The previous finding shows that phishing can be a strong attack if done correctly, thus it can be used to steal sensitive information from ignorant users of any service. This raises the question on how can one know that a website is trustable? This can be answered by answering the opposite question, what makes a fake website trustable.

The authors of [11] answered the later question, they said “Successful phishers must not only present a high credibility web presence to their victims; but they must also create a presence that is so impressive which causes the victim to fail to recognize security measures installed in web browsers”. Hence, the presence of the website is the main influence in the success of the phishing attempt. In our opinion, what makes phishing a dangerous attack is the fact that it allows the attacker to penetrate a system without going through the normal defenses.

III. Related Work

As shown above, social media has become an important platform for cyber criminals. Over the years, researchers and scientists have studied spam and phishing attacks to develop ways that will help in detecting and preventing them. None of the current techniques guarantee its results; however, some of them have achieved a tolerable percentage so that it can be cost effective to use. There is an important relationship between detecting spams and preventing them. In [2] the authors inferred that you cannot have prevention without detection by saying “Detecting spam is the first and very critical step in the battle of fighting spam”. In [12], the authors mentioned that the length of a false URL differs from the normal one and, thus, it is possible to distinguish fake URLs from the trusted ones. Moreover, in their study on the behavior of the attackers, they found that they usually misuse the webhosting services (mostly free). In addition, they claimed that the domains that become active immediately after registration is most likely associated with phishing purposes. Finally, they mentioned that it is a fact that the machines hosting the phishing domains are distributed across different countries, this proves that botnets are used in phishing attacks.

In [13], an experiment on Twitter hashtags was conducted; the authors created a hashtag on Twitter and monitored the users using it. Their observation showed that after a hashtag becomes popular spammers start using it. Furthermore, they established some features to distinguish spam accounts from genuine accounts. They claimed that the frequency of tweets between the two groups are different, as the spammers tweet with higher frequency than the legitimate users with a mean of 8.66 Tweets Per Day (TPD). On the other hand, the legitimate user tweets with a frequency of 6.7 TPD. Another feature that they found is the friend to follower ratio; they claim that the legitimate user has a higher ratio than a spammer.

In [14], a new way of detecting spam was introduced by the authors, they created 900 user accounts on three different social media websites (Twitter was one of them). They called the newly created group honey-profiles, from that point they started to log all activities in the accounts being either malicious or legitimate for a year. They stated, “Even if friend requests are unsolicited, they are not always the result of spammers who reach out. In particular, many social network users aim to increase their popularity by adding as friend’s people they do not know”. Later, they started analyzing the spam on the account and came to interesting findings, they found that the level of activity differs between spammers. They, then, categorized them into four groups: displayer, bragger, whisperer, and poster. The poster showed that it is the most effective out of the four and the displayer was the least effective. Furthermore, the authors built a tool to detect the spam activity on Twitter by working more on their insights. They focused on two groups, the bragger and the poster, as they claim that they do not require genuine profiles for detection. The first strategy is called FF-ratio; it works by comparing the number of friends the user has and the number of friend request sent by him/her.

This can be considered as a variation of the technique introduced by the authors of [13] where they compare the friend to follower ratio, but the focus here is on the request sent not to the friends the users already have. The paper also studied the similarity between messages, where they say that it is possible to detect a bot user from a legitimate human user based solely on the URLs on the message. In addition, they addressed another technique for detecting spam or phishing bots by comparing the number of friends and the messages sent. The authors finalized their work by using machine learning techniques to extract features between the spam/phishing accounts that allowed them to detect spam and phishing in real time on Twitter.

The problem with this work is that the speed of the process is not fast enough since Twitter limits the machine to only run 20,000 API calls per hour. To solve the issue, they decided to get assistance from the users of Twitter by providing them with the ability to flag (mark) tweets as spam then execute the classifier on the profiles. The advantage of this technique is that it saves time, meaning that if the spams get more inelegant, we do not need to find an alternative way; instead we can retrain the data and get even stronger detection.

In [15], the authors proposed a scheme for detecting spam; the paper was solely focused on Twitter. The authors claimed that it is possible to differentiate spam/phishing tweets from legitimate tweets in two different ways: i) account feature-based relations and ii) message feature-based scheme. This means that they rely heavily on the features they learn from existing spam. However, all these schemes are time and resource consuming; spam is a moving target and difficult to measure.

The authors of [2] introduced a new perspective for detecting spam/phishing on Twitter since their approach takes into consideration the performance factor. They elaborated on [14] by commenting that it can barely reach the near real time requirements by Twitter. The authors continued by reporting that with the increased popularity of Twitter the traditional ways that were used before the age of social media is not effective and should not be used anymore. They thought that detecting spam is not an achievement if you do not have acceptable performance rate in the system. This takes into consideration the plea of near real-time delivery where traditional techniques will consume too much computational power and will not be able to meet the time requirement. They continued describing their new approach by saying “Our work shifts the perspective from individual detection to collective detection and focuses on detecting spam campaigns”. This will increase performance dramatically since the focus will be shifted to a cluster of tweet as opposed to one tweet at a time. They proceeded on efficiency by claiming that their approach clusters related spam accounts into a campaign and generates a signature for the spammer behind the campaign. Thus, not only their work can detect multiple existing spam accounts at a given time, but it can also capture future ones if the spammer maintains the same spamming strategies. And in regard to robustness, they reported, “Twitter defines the behavior of posting duplicate content over multiple accounts as spamming. By grouping related accounts, our work can detect such a collective spamming behavior”. In our opinion, focusing on the group level is a brilliant idea and should in theory increase the performance of any given system and increase the speed of detect/prevent since spam shares some common characteristics and the future detection feature that they introduced. This makes sense because spammers promote their content in large scale campaigns as [17, 18] described.

The effectiveness of a web spam will increase if the domain associated with it is more popular around the web in particular search engines, as they are the root of finding websites on the Internet. In 2007, Microsoft started a research project to investigate web spam. They described web spam in [19] as “...Web spam refers to pages that use techniques to mislead search engines into assigning them higher rank...” from this definition, we can see that spam is giving itself more undeserved popularity to gain as much visits as possible. They found that the construction of the dataset is crucial to improve accuracy of spam classification. This relates heavily to the idea of this paper, as if the spammer on Twitter could perform web spam to increase the popularity of its domain on the web, this might earn him/her a spot on Alexa’s most visited websites worldwide. In this case, the detection algorithm will skip him/her since it is not a suspicious website anymore. Microsoft continued by categorizing the methods of increasing popularity “…There are numerous ways to improve a site’s ranking, which may be broadly categorized as ethical, or white-hat, and less ethical, or grey-hat (or black-hat), SEO techniques…”. The ethical techniques are not harmful; in fact, they might improve the sites content; the most harmful category are unethical ways. The authors of [20] talked in-depth about web spam and described several techniques organizing them into taxonomy, most importantly they concluded their paper with the fact that their taxonomy leads to some techniques that could be used by the search engine providers to fight web spam.

One year after the launch in 2006, Twitter had pushed its first update in the battle of fighting spam. They announced in [21] the start of the new admin tool as they called it; it was designed to help the support staff in dealing with spam accounts after they are detected by suspending them. In addition, they introduced the community powered alerts to help the administrators identify spam account blocked by users and suspend them. Then, Twitter hired a detected staff to deal only with spam problems. Before this update, Twitter had no spam counter measures at all.

After one year from the first spam related update, a new update was pushed. This time Twitter realized that no one could detect spam as humans, so they allowed the users to help in the process of detection by flagging tweets as spam. In [22], they reported “Today we’ve added another tool to our spam fighting toolbox that will give users the ability to flag bad accounts on Twitter”. This update was a huge step forward to how they deal with spam. Now, if the spam filter failed due to the sophistication of the spam, normal users will still act as a defense and will report the account for the admin to take action. After that, the spam can be fed to the detection system to increase its accuracy for future detections.

In 2010, Twitter started to take action against phishing attacks. They noticed that phishing is becoming more popular on the service and that there is exploitation to the trust relation in the Direct Message (DM) feature. Based on that, they were obligated to release an update that will deal with the issue. In [23] they announced that the DM system is being redesigned in a way that allows users to send/receive DMs from users they follow. They believe that this approach should reduce the number of phishing attacks.

Most lately, in continuing their fight with spam/phishing, Twitter announced in [25] the system of bot maker. It was designed to achieve the following objectives: i) preventing spam content from being created; ii) reducing visibility time of spam in Twitter; iii) reducing reaction time to new spams. The system works as follows, the distributed systems feeds events to the bot maker, and the bot maker goes through the content over a set of rules then act accordingly. The rules are grouped into two parts, condition and action. The conditions are placed to help in deciding whether it is a spam or not, while the action is what will follow the condition if it is met. In their study, the service had 40% less spam since the launch of the bot maker. Ideally spam should be detected at real-time or near real-time, however, in reality this is hard to achieve because of performance issues. The cleverness that went into the design of the bot maker is that it consists of multiple stages. The first stage is the real-time stage that should provide the system with the capability to detect spam on run time; mechanisms like CAPTCHA are placed at this stage. The second stage is the near real-time, when the first stage fails the second stage kicks in, ML is a key concept in this stage to train and classify the objects on the system. The final stage is the periodic jobs stage, this stage consists of a model that extracts features and similarities between user accounts by evaluating the user's activities over a period of time, this stage can be run off line.

To sum up, spam is a real issue that affects the user experience in social media and there are multiple research papers [26-40] aimed to fight the existence of spam. Many of them focus on social media as a broad category and since Twitter is considered a microblogging service with different user requirements, this broad category of research does not always fit to Twitter. To be as precise as possible, we have focused as much as we can on the papers that explicitly mentioned Twitter as a service. Overall, the draw-back of the current literature are usually one of two, either it is not accurate enough, or it is not fast enough. The proposed work aims to provide a solution that is accurate and fast enough to be used in near-real-time application.

IV. Outline of the Proposed Approach

The main objective of this work is to introduce and develop a new model for detecting/preventing spam messages in near real-time. The proposed approach focuses on filtering and flagging tweets based on domain popularity then analyze them using ML algorithms to extract features that can help in future spam detection. Our goal is to detect spam messages that could lead to further damage, not just general spam messages. The focus of the work will be on the URLs associated with the message itself since it is the most common way to spread malicious content on the Internet. As shown in Fig. 1, the proposed approach comprises the following phases:

1) *Collection phase:* Collecting tweets periodically; (e.g. tweets in one hour).

2) *Filtering phase:* Selecting the tweets that appears with a frequency more than a predefined threshold.

3) *Flagging phase:* Examining selected tweets via popular domains on the web and flag the potential spams.

4) *Feature extraction phase:* Running ML algorithms on flagged tweets to extract the features that could be useful in detecting spam tweets.

5) *Detection phase:* Detect spam in real time using the features learned by the ML algorithm.

In periods, tweets will be collected and filtered by selecting the tweets that appear more than the decided threshold in the specified period (i.e., common tweets). After that, an examination will be conducted on the common tweets by checking the associated URL domain with Alexa's top one million globally viewed websites. The assumption is, if a tweet is common on Twitter and does not appear on the top one million globally viewed websites (e.g. google.com), it will be flagged as a potential spam. Thus, the common tweets on Twitter, but not on Alexa's will be flagged as potential spam message. Furthermore, the proposed model is reinforced by ML techniques for feature extraction to increase detection accuracy. Therefore, after flagging the potential spam messages, ML algorithms will be run on the flagged messages to extract features that help identify the cluster of spam in the future and prevent it in real-time.

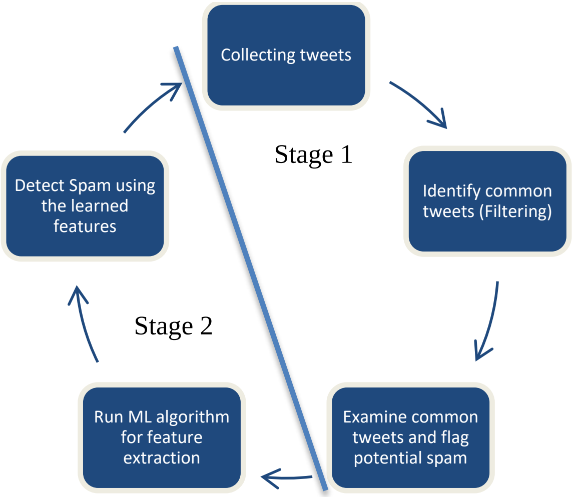
Collecting tweets
Stage 1
Identify common tweets (Filtering)
Examine common tweets and flag potential spam
Run ML algorithm for feature extraction
Stage 2
Detect Spam using the learned features

Fig. 1. The Proposed Model Outline.

V. Performance Evaluation

In this section, we describe in detail the proposed detection model, the datasets used to develop it, and the set of experiments conducted to validate it and evaluate its performance.

A. *Collection Phase*

We have collected a dataset of 27 days' (42TB) worth of tweets with a total of 268,921,568 records each record represents 1 tweet. This can be considered as the initial dataset and will be referred to as dataset1. Moreover, Spamhause is a company that collects and releases a list of confirmed spam domains, these domains will be helpful in detecting some of the false negatives results.

B. *Filtering Phase*

After collecting tweets and constructing dataset1 (step 1), the proposed model will need to filter the collected data to have a training set for the ML algorithm. The second step is to decide a threshold for the frequency of the tweets in the one-hour period; the tweets in dataset1 that have frequency exceeding this threshold are selected for popular domain test. These tweets are called common tweets. The initial value for the threshold was 120 tweets/hour (2 tweets per min); the test started with this value. Later, after few iterations of the process; manual analysis of the results showed that this value seems a bit low as the percentage of the domains that showed a frequency between 120 and 186 was benign with a 67.8%, thus the threshold was increased to 200 (3.33 tweets per min). The message that have frequency 200 or more are gathered in dataset2 and any spam messages that has a frequency bellow 200 will not be included. At the end, dataset2 had a size of 75,678,885 common tweets; among them are 19,658,349 are actual spam and 56,020,162 are not spam.

C. Flagging Phase

In the third step, the common tweets (i.e. those appear with a frequency 200 or more in a period of 1 hour) are tested via popular domains on the web; a common tweet is flagged as a potential spam if it does not appear on the top one million globally viewed websites. After applying this rule, 23,026,928 tweet messages were extracted from dataset2 in a list of 1131 distinct domain, this dataset will be referred to as dataset3. The distinct domains have been tested manually; we have visited each and every domain using a virtual machine to protect our own systems.

The confusion matrix after applying phase 3 (i.e., at the end of stage 1) is shown in Table I. In order to evaluate the performance of the first stage; we have used the performance metrics expressed in Equations 1-4. The precision was valued at 84.5% and the sensitivity is at 99%. Even though the precision is quite low, it is still incredibly good for the first stage. The performance values of stage 1 are shown in Table II.

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Sensitivity\ (recall) = \frac{TP}{TP+FN} \tag{2}$$

$$F1 - score = \ 2 \times \frac{Precision \times Sensitivity}{Precision + Senstivity} \tag{3}$$

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{4}$$

TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative.

TABLE I. CONFUSION MATRIX PRODUCED BY STAGE 1

	Actual Spam (+) 19,658,349	Actual Not Spam (-) 56,020,536
Flagged as Spam (+) 23,026,928	TP 19,461,766	FP 3,565,162
Flagged as Not Spam (-) 52,651,957	FN 196,583	TN 52,455,374

TABLE II. PERFORMANCE EVALUATION AFTER STAGE 1

Precision	Sensitivity	F1-Score	Accuracy
84.5%	99%	91%	95%

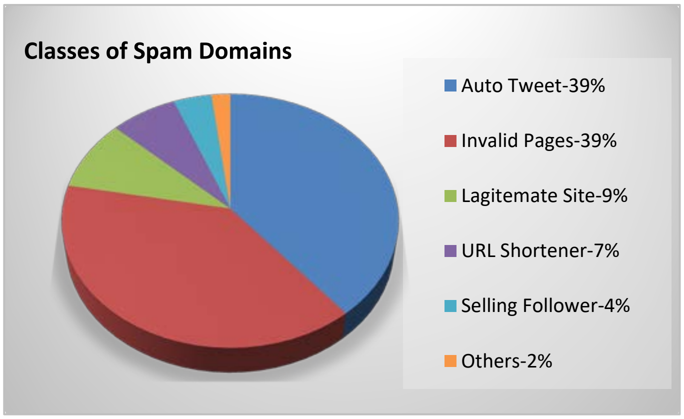

Fig. 2. Spam Domain Classification.

Fig. 2 shows the spam domain classes; we can see that highest percentage is for the invalid pages (i.e. pages that are not working anymore). This shows that spammers create their website for a specific purpose and then dispose them after it completes its objective. Another large percentage was for the tweeting services, many users on Twitter use such services to auto tweet some content they want on their timeline at specific periods.

D. Feature Extraction Phase

To extract spam features, we need to find similarities between users of the social media websites. First, we need to extract features from each user of the service; according to [26] there are two categories of features: user-based features where the focus is on the relations of the user (e.g. followers/friends) and content-based features where the focus is on the content that the users post like tweet messages. Three different classifiers (Random forest, J48 and Naïve Bayes) implemented in Waikato Environment for Knowledge Analysis (Weka), were employed in stage 2 to ensure the accuracy of the results. Weka is free software licensed under the GNU General Public License and developed at the University of Waikato, New Zealand.

1) User-based features

a) User reputation: The first feature studied is user reputation, the authors of [26] commented on user reputation by saying “Spam accounts try to follow large number of users to gain their attention”. In [27] the author mentioned that spam accounts have the tendency to follow a large amount of users to gain attention. Thus, he created the following formula to calculate the reputation, R(vi), of a single user (vi),

$$R(vi) = \frac{f(vi)}{f(vi)+o(vi)} \tag{5}$$

where f(vi) the number of friends and o(vi) the number of followers. From this formula, we can see that if the number of friends is small compared to the number of followers then the reputation will be low (i.e. close to zero) and according to the author these accounts have a high probability of being spam. The reputation study showed that auto tweeting services cannot be classified as spam even though, in principal, they could satisfy the definition of spam given in section 1 as they might be an irrelative message sent automatically; the users of these services have a high reputation score with a mean of 0.47rep. The reputation study also confirmed that the users that tweet outdated URLs are defiantly spammers; the top 3 invalid domains have scored a low mean of 0.19rep; it involved 983,273 users. Finally, the reputation also confirmed that shortened URL in general cannot be used to classify any group of users since it is used by all. Hence, the reputation helped in finding some of the domains that are used heavily by spammers due to the low reputation score such as changerion.inf and coconut.chips.jp scoring 0.21 and 0.29 respectively. In general, we think that detecting spam using reputation is an outdated technique because of shortened URL and auto tweeting services that dominate the messages on the social media.

b) User followers and friends count: In the previous feature (reputation) the focus was on the percentage given by the reputation formula, so two users could have the same reputation, but this does not necessarily mean that they have the same followers and friends count. The number of followers and friends is an important feature to distinguish different clusters of users. In order to study this feature, a random training set was chosen from the database with a size of 21K record (tweet). The legitimate users scored a mean of 253,032.7 followers and 4244.993 friends count. This makes sense due to the fact that active users and well-known icons like celebrities will have a high count of followers and a low count of friends. On the other hand, spammer have shown that they have a low mean number of followers compared to the legitimate users scoring; a mean of 4429.76 follower count and 3592.11 friend count. This comes from the fact that users usually follow back the user who followers them, so spammers exploit this habit by sending a flood of friend requests to a group of user account in the hope that some of them follow back.

c) User verification: Twitter provides a service of verifying known users such as celebrities, this feature could be used for detecting spam account since verified users are most likely legitimate users. A random set of 17171 tweets was chosen from the database as a training set, after classifying the dataset using the three different classifiers the result was as shown in Table III.

All the three classifiers (Random forest, Naïve Bayes and J48) gave the same results. The classification result shows that verified account are usually not spammers with a probability of 99.5% (i.e. if an account is verified by Twitter it has a 99.5% chance of not being a spam account). We can see that this feature is useful in detecting if an account is not spam (verified), but not the other way around. It is important to note that if the account is not verified, this does not mean it is a spam account as the results have shown 50.1% chance for this, this means that in order to detect spam we will need to add more features. This can be useful as a first filter after the flagging (Alexa comparison) to eliminate the accounts that are not relevant.

d) User listed count: In Twitter, each user has several public lists that he/she is a member of. By studying spammers on Twitter and visiting their pages, we have noticed that most of them are listed in different kinds of lists; most of them are in advertisement groups. However, for the purposes of this research, we believe that using the count of how many times a spammer has been listed is more accurate than checking the actual list itself due to the fact that the lists don’t have a standard naming system which can make each list unique. A random training set of 17540 labelled tweets was selected from the database to test the validity of the feature. The feature did prove as a useful feature for detecting spammers. The J48 classifier was able to distinguish spammers with the statistics shown in Tables IV and V.

Like the verified feature, the count of lists can be used to find spammers, but not the other way around, the classifier has classified 7288 tweets correctly as spam (TP) and 1810 tweet classified wrong (FP). Even though the performance is not high it is still an acceptable feature and can be added to the overall classifier.

e) User statuses count: Every post on Twitter is counted as a status of the user, this means that if a user tweets or retweets or even replies publicly to another user, the counter will count every instance. Obviously, spammers will have high statuses associated with their accounts. Hence, old and active user accounts (aka veterans) will still have a high count as well, so this feature needs to be tested by a classifier to check if it is an acceptable feature. A random training set of 26986 tweets was selected from the database for testing feature by classifiers. The first classifier (Random forest) gave expected results with good statistics; this is shown in Tables VI and VII.

The second classifier, J48, classified 12939 tweets as a spam correctly (TP) and only 1033 was classified as spam wrongly (FP). This shows that the feature (statuses count) could be considered as a strong feature to add to the final classifier. The results given are unexpected as it was stronger in detecting the legitimate tweets rather than the spam tweets. These results are shown in Tables VIII and IX. As for the Naïve Bayes classifier, the results are more in favor of detecting spam tweets not the other way around as shown Tables X and XI. However, it is clearly shown that the feature is unreliable since 88.97% of the results are classified as spam. Finally, this feature has shown verity in the result in all the classifier. Except for Naïve Bayes classifier, the feature is valuable and can be used in the final classifier to distinguish between the two classes.

TABLE III. CONFUSION MATRIX (USER VERIFICATION FEATURE)

	Spam (+)	**Not Spam (-)**
Not verified (+)	TP=5001	FP=4978
Verified (-)	FN=38	TN=7154

TABLE IV. CONFUSION MATRIX BY J48 (LISTED COUNT FEATURE)

	Spam (+)	**Not Spam (-)**
Classified as spam (+)	TP=7288	FP=1810
Classified as not spam (-)	FN=2564	TN=5878

TABLE V. PERFORMANCE EVALUATION BY J48 (LISTED COUNT FEATURE)

Precision	Sensitivity	F-Measure	Accuracy
0.801	0.74	0.769	0.751

TABLE VI. CONFUSION MATRIX BY RANDOM FOREST (STATUSES COUNT FEATURE)

	Spam	***Not spam***
Classified as spam	13321	651
Classified as not spam	2325	10689

TABLE VII. PERFORMANCE EVALUATION BY RANDOM FOREST (STATUSES COUNT FEATURE)

Precision	Sensitivity	F-Measure	Accuracy
0.9534	0.8514	0.8995	0.8897

TABLE VIII. CONFUSION MATRIX BY J48 (STATUSES COUNT FEATURE)

	Spam	***Not spam***
Classified as spam	12939	1033
Classified as not spam	171	12843

TABLE IX. PERFORMANCE EVALUATION BY J48 (STATUSES COUNT FEATURE)

Precision	Sensitivity	F-Measure	Accuracy
0.9261	0.9869	0.9555	0.9553

TABLE X. CONFUSION MATRIX BY NAIVE (STATUSES COUNT FEATURE)

	Spam	***Not spam***
Classified as spam	13397	10612
Classified as not spam	1030	1947

TABLE XI. PERFORMANCE EVALUATION OF NAIVE (STATUSES COUNT FEATURE)

Precision	Sensitivity	F-Measure	Accuracy
0.5579	0.9286	0.6970	0.5685

f) User favorite count: Users of tweets can flag tweets they like as a favorite, this allows users to group the messages they like and view them at any time. This feature does not only benefit the user him/herself, but other users can go into his/her account and check out his favorite list. The user's favorite count can be used as a feature in the classifier to detect spammers. Similar to other features, a proper testing has been conducted on a training set to check if the feature is acceptable in distinguishing spammers from legitimate users. The three classifiers have been used to test the feature. Random forest and J48 classifiers has shown that spammers do not have a favorite list associated with them (i.e. the count of the list is zero), this makes sense since spammers do not care about other tweets and most of them are bots. Our first thought was; this would not be a proper feature since many users do not use the favorite flag at all. However, the classifiers have shown that this is indeed a reliable feature, not for detecting spammers but for detecting legitimate users with a sensitivity of 0.962.

2) Content-based features

a) Number of hashtags: Hashtags are features used by users of Twitter in order to group relevant tweets together. This feature introduces an opportunity for spammers to spread their content to all the users without mentioning them directly. A careful inspection of the dataset has revealed that a high appearance of hashtags is most likely associated with a spam message. A 15K random tweets were plugged in the classifiers, which gave the results listed in Tables XII and XIII.

TABLE XII. CONFUSION MATRIX (NUMBER OF HASHTAGS FEATURE)

	Spam	*Not spam*
Classified as spam	8740	5288
Classified as not spam	9	963

TABLE XIII. PERFORMANCE EVALUATION (NUMBER OF HASHTAGS FEATURE)

Precision	Sensitivity	F-Measure	Accuracy
0.6230	0.9988	0.767	0.4468

The statistical summary above is for the dataset that includes 4 or more hashtags. This shows incredibly good results with a sensitivity of 0.99. Hence, it could be considered as a reliable way for detecting spam accounts, but not the other way around.

b) Number of Mentions: In Twitter the users have the option to mention other users in their tweets by adding the (@) sign before his/her username. Spammers can use this feature to mention as much users as they can to spread their content directly to them. The classifiers have shown similar result to the number of hashtags shown above. The higher the count of mentions the more likely it is a spam; the classification gave sensitivity of 81% for messages that includes four or more mention to be spam.

c) Sensitivity of tweets: The sensitivity field in the tweet record is a Boolean field (true, false) that is only available when a tweet has a link associated with it. Obviously, this makes sensitivity feature seems to be relevant to our work since our main focus is on domain popularity (i.e. all the tested tweets must have domains associated with them). The denotation of this feature does not describe the content of the tweet itself, but in its place, it is used as an indicator that the hyperlink associated with the message may contain content identified as sensitive. This shows that the sensitivity feature may be used for classification because of its relevancy to the main idea. However, after testing a random training set of 26K tweets, the three classifiers have shown that this feature does not help in identifying spam at all. Fig. 3 shows that the sensitivity of the URL has no effect on the message being a spam or a legitimate tweet since spam tweets are not targeting one specific type of content. Thus, this feature has been dropped from the final classifier.

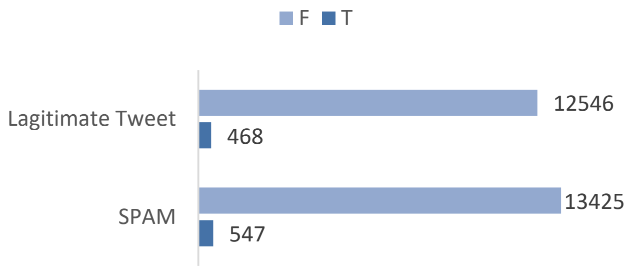

Fig. 3. Tweet Sensitivity.

E. Spam Detction Phase

After selecting the proper features, now it is time to put them together and evaluate the spam detection model. Just like the individual test for features, the same three classifiers (Ransom forest, J84 and Naïve Bayes) were used to evaluate the final spam detection model. To get the highest accuracy possible, two methods for evaluation were considered. The first method (raw record method) involves building a training set of raw tweets; this means that the labelled tweets are taken as they are from the database directly and plugged in the classifier. The second method (grouped record method) groups the tweets that advertise the same domain into one record using mean and standard deviation for each feature. Both methods have the potential to be accurate. In the first method, ML algorithms work better with raw data as they do the calculation and the classification more accurately, while in the second method each set has anomalies that may change the result; by aggregating the anomaly list, anomalies will be removed or have virtually no impact on the dataset.

Each evaluation method involves two test options: 1) in the first option (percentage split test), half of the records (tweets) are used as training set to see how well the classifier can distinguish between them and then the other half are utilized as a test set to examine how well the classifier works on unlabeled data; 2) the second test option (cross-validation test) divides the database into 10 folds, iteratively runs training on 9 folds and leaves one for testing; the test fold is changed in each iteration. This is considered an accurate method because if a fold is used for training it is not used for testing.

1) Evaluation using raw record method: A database of size 26,986 records was constructed to include 50% spam and 50% legitimate tweets, the tweets where selected at random from each type, For the purpose of the first test option (% split) the dataset has been split into two subsets, the first half to be used as a training set and the other half as test set. Next, for the second test option (cross validation) the complete dataset will be plugged into the classifiers with its entirety, the algorithm splits the dataset into ten equal folds then work on them accordingly. The evaluation process is accomplished in 10 iterations; in each iteration nine folds are chosen for training and one is left for testing; the testing fold is changed in each iteration. The records will contain 11 features (user verification, user followers count, user friends count, user reputation, user listed count, user statuses count, user favorites count, user language, tweet language, number of mentions and number of hashtags) plus the label (spam or benign).

a) Percentage split test: The Random Forest classifier was able to perfectly distinguish between the 13,493 instances giving us perfect results for the training dataset as shown in Table XIV. Unlike the random forest classifier, the J48 and Naïve Bayes could not perfectly distinguish between the two classes. However, the result is still on the good side with J48 coming second and Naïve Bayes as the worst out of the three. This shows that the features are in fact good features and they can be used to detect spam messages.

After classifiers learned how to distinguish between spam and legitimate tweets in the training stage, the other half of the dataset (test set) is used to test how well the classifiers can distinguish between unlabeled new data with the model built from the training data in the training stage. The details of the performance are shown in Table XV.

Table XV shows that random forest and the J48 classifiers have built a strong model for detecting spam that could be relied on with 0.981 and 0.946 precision and an accuracy of 92.9% and 92.5%, respectively. On the other hand, the Naive Bayes classifier had the highest and almost perfect precision with 0.988, however, the model also classified falsely nearly half of the classified set which affected its accuracy to be only 76.2%. Thus, the naïve Bayes classifier is not a reliable method of classifying such data and is not recommended to be used.

The confusion matrix in Table XVI explains the above statistics in terms of number of records each classifier has predicted correctly or wrongly. Random forests were able to classify 6877 tweets as spam correctly and only 136 tweets were classified as false positive. On the other hand, the classifier failed to classify 816 tweets that we can call false negatives. Next, the J48 classifier also have some good results, the model classified 6634 tweets correctly as spam and only 379 false positive. In contrast 257 tweets were flagged falsely as legitimate (benign) tweets. Finally, the Naïve Bayes gave a surprising result of 6931 tweets classified correctly as spam and only 82 false positive. Yet, even though Naïve Bayes showed particularly good numbers in detecting spam, it still has a high number of false negatives with 3129 records that should be flagged as spam. Therefore, we can say from the results above, that tree-based classifiers such as J48 and random forest, work very well and are accurate enough to call them valid spam detection techniques.

TABLE XIV. PERFORMANCE EVALUATION FOR SPAM CLASS (TRAINING)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	1	1	1	1
J48	0.972	0.947	0.959	0.969
Naïve Bayes	0.794	0.989	0.880	0.892

TABLE XV. PERFORMANCE EVALUATION FOR SPAM CLASS (TESTING)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	0.981	0.894	0.935	0.929
J48	0.946	0.963	0.954	0.925
Naïve Bayes	0.988	0.689	0.812	0.762

TABLE XVI. CONFUSION MATRIX FOR THE THREE CLASSIFIERS (TESTING)

Classifier		Actual Spam	Actual Legitimate
Random Forest	Classified as spam	***6877***	136
	Classified as legitimate	816	***5664***
J48	Classified as spam	***6634***	379
	Classified as legitimate	257	***6223***
Naïve Bayes	Classified as spam	***6931***	82
	Classified as legitimate	3129	***3351***

TABLE XVII. PERFORMANCE EVALUATION FOR SPAM CLASS (CROSS VALIDATION)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	0.981	0.883	0.929	0.9257
J48	0.954	0.965	0.96	0.9599
Naïve Bayes	0.989	0.689	0.812	0.8407

b) Cross validation test: Before concluding the first method (raw record), another test option will be run to verify and ensure the results. For this option, the same dataset of 26K tweets has been used. The results in Table XVII have shown uniformity with the first option test with the following detailed accuracy. The numbers are almost identical with a slight increase in the cross validation option. This makes sense because in cross validation the 10-fold option increases the variety of the training and test records.

2) Evaluation using grouped record method: In this method, the idea is to group all the tweets that advertise the same domain into one record using the mean and the standard deviation for the numeric fields and count of distinct values for the other types. The dataset contains 630 labelled grouped record (50% spam) each record represents one domain. To create the grouped record, 1000 tweets have been chosen randomly from the tweets that advertise the same domain and the record was built according to the aggregate values of the tweets. Each record will contain 21 features named: count of verified, count of not verified, mean user followers count, STD user followers, mean user friends count, STD user friend, mean reputation, STD reputation, mean user listed count, STD listed, mean user statuses count, STD status, mean user favorites count, STD user favorites, count tweet possibly sensitive, count tweet possibly not sensitive, domain of URL, mean number of mentions, STD mentions, mean number of hashtags and STD hashtags.

a) Percentage split test: Similar to the first method (single record) the random forest was able again to distinguish between the classes perfectly in the training dataset. The J48 has a decrease in sensitivity; only 0.873 and a slight decrease in the precision to 0.975. Finally, the Naïve Bayes has more sensitivity 0.958 than J48 but with the least accuracy 88.91%; this is shown in Table XVIII. With the test dataset, random forest and J48 gave very good accuracy while Naïve Bayes recorded relatively low accuracy as shown in Table XIX. This again shows that tree-based classifiers are accurate enough to call them valid spam detection techniques.

b) Cross validation test: The cross-validation check will be conducted on the entire list which have been plugged into each classifier. The test gave the results shown in Table XX. As expected, and similar to the first method, the results of the cross-validation test are quite similar to the percentage split test.

3) Comparison: Firstly, the two test options (percentage split and cross validation) have shown similar results, which are expected due to the similarity in which how each one of them work as explained previously. However, even though the difference can be considered negligible, we believe that cross validation is the more reliable option to choose in our work.

Secondly, the three classifiers (Random forest, J48 and Naïve Bayes) have been used in a variety of tests and with the same data plugged into them. The tree-based classifiers (Random Forest and J48) have shown better results in this kind of setup with random forest being better in building a solid classification model to distinguish between the classes. While on the other hand, the Bayes based classifier (Naïve) has shown the tendency to group most of the record in one class (usually spam class), this made the classifier unreliable and not recommended to be used with a similar environment.

Lastly, similarly to the two test options, the two method of presenting the data (single and grouped) gave similar results. In the comparison between the two methods, only the random forest classifier will be considered since it has shown that it is the most suitable in this setup. To begin with, both methods scored a perfect score with the training stage meaning that the learning algorithm was able to build a classification model that distinguishes spam and legitimate tweets perfectly using the training data in both methods. However, in the testing stage the single (raw) record approach performed better in terms of all evaluation metrics (precision, sensitivity, f-measure, and accuracy).

To sum up, the proposed approach has been validated through three different classifiers with the random forest classifier being the most reliable one in detecting malicious spam using domain popularity in a micro blogging environment like Twitter. The two evaluation approaches showed very similar results with the single record approach being the favored and more accurate. Cross validation with 10 folds is the most suitable test option for this work. This comparison is shown in Fig. 4.

TABLE XVIII.PERFORMANCE EVALUATION FOR SPAM CLASS (TRAINING - GROUPED)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	1	1	1	1
J48	0.975	0.873	0.921	0.9158
Naïve Bayes	0.801	0.958	0.873	0.8891

TABLE XIX. PERFORMANCE EVALUATION FOR SPAM CLASS (TESTING - GROUPED)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	0.993	0.885	0.936	0.933
J48	0.98	0.903	0.94	0.938
Naïve Bayes	0.798	0.933	0.86	0.869

TABLE XX. DETAILED ACCURACY FOR SPAM CLASS (CROSS VALIDATION - GROUPED)

Classifier	Precision	Sensitivity	F-Measure	Accuracy
Random Forest	0.958	0.876	0.915	0.911
J48	0.962	0.873	0.915	0.911
Naïve Bayes	0.797	0.958	0.87	0.880

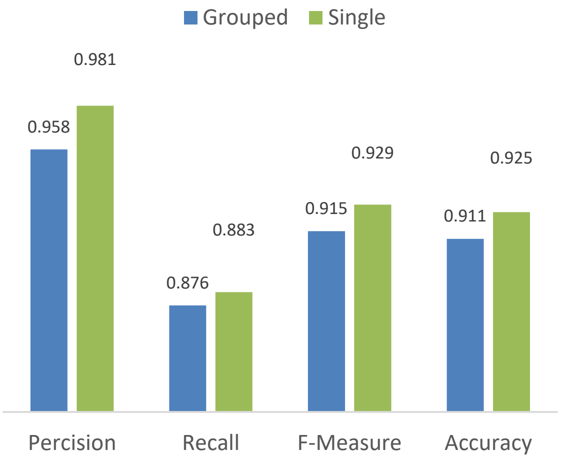

Fig. 4. Comparison between the Single and the Grouped Approach for (Random Forest - Cross Validation).

VI. Discussion

A. Operational Systems Accuracy

In this section, we discuss how Twitter may operate such a system and how accurate will it be in a real operational environment. The operation part will be the same as discussed in this paper by running the system in periodic time and performing the two stages of filtering and evaluation. However, what is not mentioned above is that the accuracy of the system will be much higher than the numbers shown in the evaluation part because of the changes between the testing environment and the real operational environment. Even though the results were good and promising in testing environment, the accuracy will be higher in the actual operation environment since the data considered in this system only referees to tweets with URLs associated with them. Thus, if the whole database was considered in the evaluation part, the numbers will be much more accurate and most likely will jump dramatically since only 8.5% of dataset 1 contains URLs associated with them. This exclusion was a necessary step to increase the accuracy as much as possible making all the records on the proposed system a possible spam message.

B. Limitations

The system may be evaded by some techniques that we can consider as limitations. The first limitation is using URL shortened, this limitation is a very powerful way to make this system useless since the spammer can mask his/her URL into another short URL using URL shorten services. However, it can be easily dealt with if the service that uses this system checks the URLs and gets the complete URL before posting the tweet and saving it as meta data in the record itself. Twitter is not vulnerable for this kind of limitation because they do in fact check the complete path of the URLs. The other limitation is auto tweeting services, in theory, these services are spreading spam since the message is going automatically from the user profile in specific times. However, one could argue that since the user is registered with them and the tweets are not random it is not a spam. The problem is that those tweets will have the domain of the tweeting service which will obviously be a popular domain in Twitter since all the users registered to those services are tweeting their domain, on the other hand, the tweeting service will most likely not be in the Alexa top visited domains. This will make the system flag those service as potential spam even though most of its users are legitimate users.

VII. Conclusions

History has shown that the fight against spam is a cat and mouse game, it is a never-ending battle. Whenever a countermeasure is introduced spammers find a way to evade it. Though, history have also shown that instead of trying to defeat spam entirely we should focus on reducing it to an acceptable rate. In this paper, we have introduced a new way of detecting malicious spam that has never been used before in popular online micro blogging services, focusing mainly on Twitter service. The problem in hand is not to detect spam in general, but to detect malicious spam that could escalate to other type of attacks. Thus, the idea focuses on URLs associated with the messages since they are the most common way used to spread malicious content. Based on Twitter spam policy, the content-based and the user-based features are used in ML algorithm to detect spam messages. This work has added a filtering stage before the ML stage to increase the efficiency and accuracy of detecting such spam type. Furthermore, three different classification algorithms have been studied and used to analyses the data. The results show that filtering the popular domains that appear in Alexa’s top one million most visited websites to separate the potential spam before using the ML algorithms is a valid and accurate approach. In Addition, the classifiers were able to identify the similarities between spam messages which allows for future real time detection.

References

[1] “Top Sites,” [Online]. Available: http://www.alexa.com/topsites. [Accessed 28/02/2015].

[2] Z. Chu, I. Widjaja and H. Wang, “Detecting Social Spam Campaigns on Twitter,” Applied Cryptography and Network Security, Springer Berlin Heidelberg, pp. 455-472, 2012.

[3] H. Nguyen, “State of social media spam,” Nexgate, San Francisco, California, 2013.

[4] “Oxford Dictionary,” [Online]. Available: http://www.oxforddictionaries.com/us/definition/american_english/spam [Accessed 18/02/2015].

[5] J. M. Rao and D. H. Reiley, “The Economics of Spam,” Journal of Economic Perspectives, vol. 26, no. 3, pp. 87-110, 2012.

[6] L. F. Cranor and B. A. LaMacchia, “Spam!,” Communications. ACM, vol. 41, no. 8, pp. 74-83, Aug. 1998.

[7] M. Sahami, S. Dumais, D. Heckermany and E. Horvitzy, “A Bayesian Approach to Filtering Junk E-Mail,” AAAI-98 Workshop on Learning for Text Categorization, Madison, WI, 1998.

[8] D. Wang, S. B. Navathe, l. liu, D. Irani, A. Tamersoy and C. Pu, “Click Traffic Analysis of Short URL Spam on Twitter,” The 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, Austin, TX, USA, 2013.

[9] C. Griery, K. Thomas, V. Paxson and M. Zhang, “@spam: The Underground on 140 Characters or Less,” 17th ACM conference on Computer and communications security, New York, NY, USA, 2010.

[10] R. Basnet, S. Mukkamala and A. H. Sung, “Detection of Phishing Attacks: A Machine Learning approch,” in Soft Computing Applications in Industry, Berlin, Germany, Springer Berlin Heidelberg, pp. 373-383, 2008.

[11] R. Dhamija, J. D. Tygar and M. Hearst, “Why Phishing Works,” Human Factors in Computing Systems (CHI), Montr´eal, Canada, 2006.

[12] M. D. Kevin and G. Minaxi , “Behind Phishing: An Examination of Phisher Modi Operandi,” San Francisco, CA, 2008.

[13] S. Yardi, D. M. Romero, G. Schoenebeck and d. boyd, “Detecting spam in twitter network,” Firstmonday, vol. 15, no. 1, 2009.

[14] G. Stringhini, C. Kruegel and G. Vigna, “Detecting Spammers on Social Networks,” The 26th Annual Computer Security Applications Conference(ACSAC), New York, NY, USA, 2010.

[15] C. Zi , G. Steven, H. Wang and S. Jajodia, “Who is Tweeting on Twitter: Human, Bot, or Cyborg?,” The 26th Annual Computer Security Applications Conference, New York, NY, USA, 2010.

[16] H. . D. Nannaware., T. . P. Dhangar., A. S. Dhanrao. and P. V. D. Badgujar., “A Run-time Detection System for Malicious URLs in Twitter,” International Journal on Recent and Innovation Trends in Computing and Communication, vol. 3, no. 1, pp. 144 - 147, 2015.

[17] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen and B. Z, “Detecting and Characterizing Social Spam Campaigns,” IMC '10 Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, New York, 2010.

[18] M. Egele , G. Stringhini, C. Kruegel and G. Vigna, “COMPA: Detecting Compromised Accounts on Social Networks,” NDSS symposium, San Diego, CA, USA, 2013.

[19] K. M. Svore, Q. Wu and C.. J. Burges, “Improving web spam classification using rank-time features,” Proceedings of the 3rd international workshop, New york, 2007.

[20] Z. Gyongyi and H. Garcia-Molina, “Web Spam Taxonomy,” First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb 2005), Japan, 2005.

[21] “Turning Up The Heat On Spam,” Twitter, 21 Aug 2008. [Online]. Available: https://blog.twitter.com/2008/turning-heat-spam. [Accessed 2/ 4/2015].

[22] “Help us nail pammers,” Twitter, 13 Oct 2009. [Online]. Available: https://blog.twitter.com/2009/help-us-nail-spammers. [Accessed 2/4/ 2015].

[23] “Avoid phishing scams,” Twitter, 26 Feb 2010. [Online]. Available: https://blog.twitter.com/2010/avoid-phishing-scams. [Accessed 2/4/ 2015].

[24] “State twitter spam,” Twitter, 23 Mar 2010. [Online]. Available: https://blog.twitter.com/2010/state-twitter-spam. [Accessed 4/4/2015].

[25] “Fighting spam with botmaker,” Twitter, 20 Aug 2014. [Online]. Available: https://blog.twitter.com/2014/fighting-spam-with-botmaker. [Accessed 4/4/2015].

[26] M. McCord and M. Chuah, “Spam Detection on Twitter Using Traditional Classifiers,” Proceedings of the 8th international conference on Autonomic and Trusted Computing (ATC'11), Berlin, 2011.

[27] A. H. Wang, “Don’t follow me: Spam Detection in Twitter,” Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on, Athens, Greece, 2010.

[28] M. P. S. Michael Gilleland, “Levenshtein Distance, in Three Flavors,” university of pittburgh, [Online]. Available: http://people.cs.pitt.edu/~kirk/cs1501/Pruhs/Fall2006/Assignments/editdistance/Levenshtein%20Distance.htm. [Accessed 13/7/2015].

[29] R. J. R. Raj, S. Srinivasulu, and A. Ashutosh, “A Multi-classifier Framework for Detecting Spam and Fake Spam Messages in Twitter,” 2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT), Apr. 2020.

[30] N. Imam and V. Vassilakis, “Detecting Spam Images with Embedded Arabic Text in Twitter,” 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), Sep. 2019.

[31] Rajasekaran Rajkumar and Jolly Masih, “Opinion Analysis on Twitter Data and Detecting Spam Tweets,” International Journal of Innovative Tech. and Exploring Eng., vol. 9, no. 2, pp. 711–714, Dec. 2019.

[32] O. Çıtlak, M. Dörterler, and İ. A. Doğru, “A survey on detecting spam accounts on Twitter network,” Social Network Analysis and Mining, vol. 9, no. 1, Jul. 2019.

[33] M. Mostafa, A. Abdelwahab, and H. M. Sayed, “Detecting spam campaign in twitter with semantic similarity,” Journal of Physics: Conference Series, vol. 1447, p. 012044, Jan. 2020.

[34] K Subba Reddy and E. Srinivasa Reddy, “Detecting Spam Messages in Twitter Data by Machine Learning Algorithms using Cross Validation,” International Journal of Innovative Tech. and Exploring Eng., vol. 8, no. 12, pp. 2941–2946, Oct. 2019.

[35] S. Linganur, “Detecting Spam in Twitter and Email using Machine Learning Approach,” International Journal for Research in Applied Science and Engineering Technology, vol. 7, no. 5, pp. 1652–1655, May 2019.

[36] N. Senthil Murugan and G. Usha Devi, “Detecting Streaming of Twitter Spam Using Hybrid Method,” Wireless Personal Communications, vol. 103, no. 2, pp. 1353–1374, Feb. 2018.

[37] Z. Alom, B. Carminati, and E. Ferrari, “Detecting Spam Accounts on Twitter,” 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Aug. 2018.

[38] X. Zhang, Z. Li, S. Zhu, and W. Liang, “Detecting Spam and Promoting Campaigns in Twitter,” ACM Transactions on the Web, vol. 10, no. 1, pp. 1–28, Feb. 2016.

[39] N. Eshraqi, M. Jalali, and M. H. Moattar, “Detecting spam tweets in Twitter using a data stream clustering algorithm,” 2015 International Congress on Technology, Communication and Knowledge (ICTCK), Nov. 2015.

[40] D. Sipahi, G. Dalkılıç, and M. H. Özcanhan, “Detecting spam through their sender policy framework records,” Security and Communication Networks, vol. 8, no. 18, pp. 3555–3563, May 2015.