6 Million Spam Tweets: A Large Ground Truth for Timely Twitter Spam Detection

Chao Chen*, Jun Zhang*† Xiao Chen*, Yang Xiang* and Wanlei Zhou*
*School of Information Technology
Deakin University, Victoria 3125, Australia
Email: {chao.chen, jun.zhang, xiao.chen, yang.xiang, wanlei.zhou}@deakin.edu.au
† School of Computer and Information Science
Southwest University, Chongqing 400715, P. R. China

***Abstract*—Twitter has changed the way of communication and getting news for people's daily life in recent years. Meanwhile, due to the popularity of Twitter, it also becomes a main target for spamming activities. In order to stop spammers, Twitter is using Google SafeBrowsing to detect and block spam links. Despite that blacklists can block malicious URLs embedded in tweets, their lagging time hinders the ability to protect users in real-time. Thus, researchers begin to apply different machine learning algorithms to detect Twitter spam. However, there is no comprehensive evaluation on each algorithms' performance for real-time Twitter spam detection due to the lack of large groundtruth. To carry out a thorough evaluation, we collected a large dataset of over 600 million public tweets. We further labelled around 6.5 million spam tweets and extracted 12 light-weight features, which can be used for online detection. In addition, we have conducted a number of experiments on six machine learning algorithms under various conditions to better understand their effectiveness and weakness for timely Twitter spam detection. We will make our labelled dataset for researchers who are interested in validating or extending our work.**

I. Introduction

Online Social Networks (OSNs), like Twitter and Facebook, have become integral to people's daily life in the last few years. Users spend vast time in OSNs making friends with people who they are familiar with or interested in. After the relation is built, users can view messages, usually something interesting or recent activities shared by friends they are connected to, in the terms of tweets, wall posts or status updates. Twitter, which was founded in 2006, has become one of the most popular microblogging service sites. Nowadays, 200 million Twitter users generate over 400 million new tweets per day [1].

Due to the increasing popularity of Twitter, spammers are turning into the fast-growing platform. Twitter spam, which is referred as unsolicited tweets containing malicious links that directs victims to external sites containing malware downloads, phishing, drug sales, or scams, *etc* [2], has already affected a number of users. In April of 2014, Twitter was flooded with an avalanche of spam tweets that were sent by loads of compromised accounts [3].

As a result, the research community, as well as Twitter itself, has proposed a number of spam detection schemes to make Twitter a spam-free platform. For instance, Twitter has applied some rules to suspend accounts if they behave abnormally. Those accounts, which are frequently requesting to be friends with others, sending duplicate contents, mentioning other users or posting URL-only contents, will be suspended by Twitter [4]. At the same time, researchers have proposed innovative mechanisms to detect Twitter spam [2], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15].

While some preliminary works use heuristic rules to detect Twitter spam [14], [15], most of the other works make use of machine learning algorithms. Some of them [2], [7], [10], [12], [13] are using account based features, while others [2], [5], [8] are using message based features. Account based features include the number of followers or friends, the ratio of tweets which contain URL, and account age, *etc.* As these features are easily to be evaded by buying more followers from the underground market or by other means, Yang [7] has proposed more robust features which rely on the neighbours of the user or the whole social network, such as Local Clustering Co-efficiency and Betweenness Centrality; while in [10], distance and connectivity were proposed. However, these features are very expensive to be calculated in the terms of time and computing resources, which makes online detection become nearly impossible. On the other hand, message based features are extracted from tweets' content, such as the length of a tweet, the numeric characters in a tweet and some special marks (like @ and #) in a tweet. More specifically, some researchers are making use of the URLs embedded in tweets to detect spam, *e.g.*, [5] and [8].

Despite many existing algorithms for Twitter spam detection, there is no comprehensive evaluation of different machine learning algorithms' performance. This remains a challenge as the lack of publicly available large dataset setting as a benchmark to evaluate algorithms' performance under different scenarios.

To calibrate the effort of machine learning based Twitter spam detection, we have conducted a comprehensive evaluation on six algorithms: *Random Forest, C4.5 Decision Tree, Bayes Network, Naive Bayes, k Nearest Neighbour, and Support Vector Machine.* Firstly, we have collected a huge amount of more than 600 million public tweets. Then we further labelled 6.5 million spam tweets and 6 million non-spam tweets by using Trend Micro's Web Reputation Service. We also extracted some lightweight features which are capable for timely spam detection and performed a number of experiments

978-1-4673-6432-4/15/$31.00 ©2015 IEEE

Authorized licensed use limited to: INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.

of detecting Twitter spam using our dataset. We will also make this groundtruth available on our website [1].

II. Related Work

The severe spam problem in Twitter has already drawn researchers' attention. They have proposed a number of ways to tackle this problem. Some preliminary works used heuristic rules to detect Twitter spam. [14] used a simple algorithm to detect spam in #robotpickupline (the hashtage was created by themselves) by using three methods: suspicious URL searching, matching username pattern and keyword detection. [15] simply removed all the tweets which contained more than three hashtags to filter spam in their dataset to eliminate the impact of spam for their research.

Other works applied machine learning algorithms for Twitter spam detection. [2], [12], [13] made use of account and content features, such as account age, the number of followers/followings, URL ratio and the length of tweet, to distinguish spammers and non-spammers. These features can be extracted efficiently but also fabricated easily. Thus, some works [10], [7] proposed robust features which rely on the social graph to avoid feature fabrication. Song *et al.* extracted the distance and connectivity between a tweet sender and its receiver to determine whether the tweet is spam or not [10]. While in [7], Yang *et al.* proposed more robust features based on the social graph, such as Local Clustering Coefficient, Betweenness Centrality and Bidirectional Links Ratio. Such features were proved to be more discriminative than the features in previous works. However, collecting these features are very time-consuming and resource-consuming, as the Twitter social graph is extremely huge. Consequently, these features are not suitable for online detection.

Instead, [5] and [8] solely relied on the embedded URLs in tweets to detect spam. A number of URL based features were used by [5], such as the Domain tokens, path tokens and query parameters of the URL, along with some features from the landing page, DNS information, and domain information. In [8], the authors studied the characteristics of Correlated URL Redirect Chains, and further collected relevant features, like URL redirect chain length, Relative number of different initial URLs, *etc.* These features also show their discriminative power when used for classifying spam. However, these two schemes can only detect spam with URLs, as pointed out by a recent work [11]. The systems will miss the spam with only text or fabricated URLs. [11], thus, proposed a model based spam detection scheme. They built several models, like Language model and Posting Time model, for each user. Once the model behaved abnormally, there would be a compromise of this account, and this account was likely to be used to spread spam by attackers. This method can detect whether an account was compromised or not, but cannot determine the accounts which were created by spammers fraudulently.

[1] http://anss.org.au/nsclab/resources

III. A large data set of spam tweets

A. Collection Procedure

We used Twitter's Streaming API to collect tweets with URLs. The public Streaming APIs can get 1% of all the public tweets flowing through Twitter. While it is possible to use Twitter to send spam and other messages without using URLs, the majority of spam and other malicious messages on the Twitter platform contain URLs [11]. In the thousands of spam tweets which were manually inspected, we found only a handful of tweets without URLs which could be considered as spam. In addition, spammers mainly use embedded URLs to make it more convenient to direct victims to their external sites to achieve their goals, such as phishing, scams, and malware downloading [16]. Therefore, we restricted this research to the tweets with URLs. During the collection period, we collected a total of over 600 million tweets with URLs.

B. Ground Truth

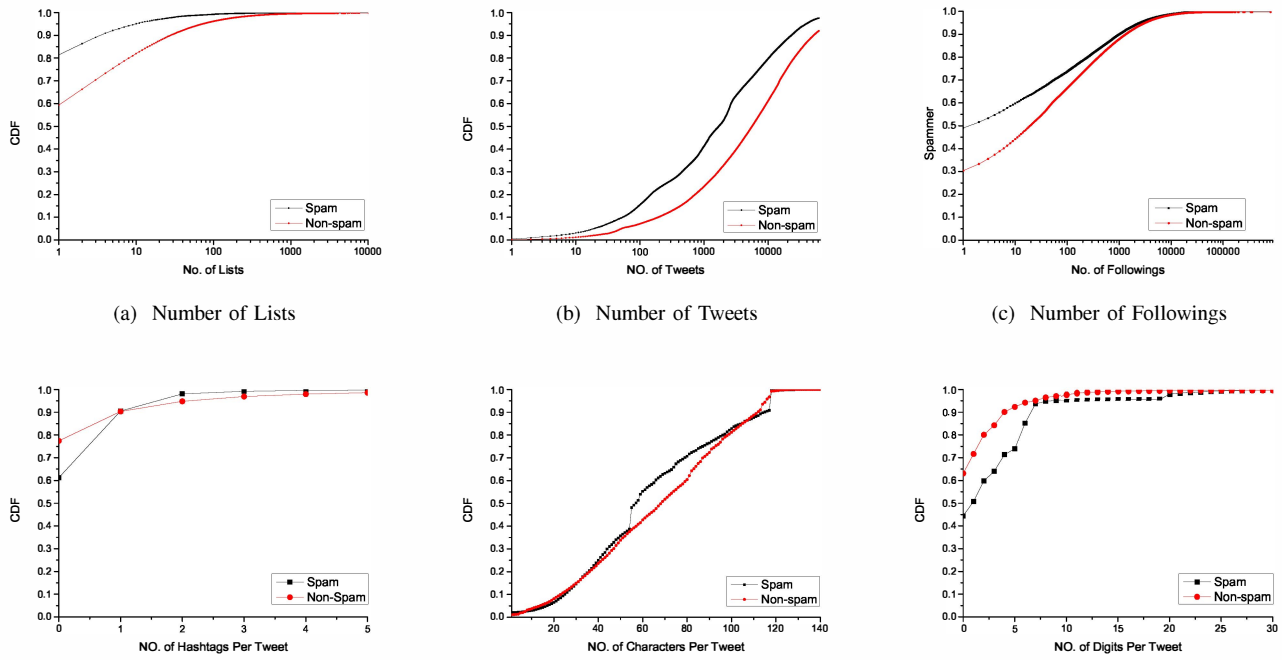
Currently researchers are using two ways to generate groundtruth, manual inspection and blacklists filtering. While manual inspection can label a small amount of training data, it is very time- and resource-consuming. A large group of people is needed during the process. Although HIT (human intelligence task) websites can help to label the tweets, it is also costly, and sometimes the results are doubtful [17]. Others apply existing blacklisting service, such as Google SafeBrowsing, to label spam tweets. Nevertheless, these services' API limits make it impossible to label a large amount of tweets.

We used Trend Micro's Web Reputation Service [18] to identify which URLs were deemed malicious tweets. Trend Micro's WRS maintains a large dataset of URL reputation records, which are derived from Trend Micro customer opt-in URL filtering records. WRS is dedicated to collecting the latest and the most popular URLs, to analysing them, and then to providing Trend Micro customers with real-time protection while they are surfing the Internet. Hence, through checking URLs with the WRS service, we are able to identify whether a URL is malicious and the category a URL belongs to. We define the tweets which contain malicious URLs as Twitter spam. In our dataset of 600 million tweets, we identified 6.5 million malicious tweets, which accounted for approximately 1% of all tweets.

C. Light-weight Statistical Features

After labelling the spam tweets, we further extracted the features from them. Since Twitter's Public Streaming API only returned random public tweets and they were not socially connected, we were not able to build a social graph from the data. As a result, it is unrealistic for us to extract social graph based features such as Local Clustering Coefficient, Betweenness Centrality [7] and distance [10]. In addition, it is significantly expensive to collect these features, as they cannot be calculated until the social graph is formed. As a result, those expensive features are not suitable to be used in real-time detection, despite that they have more discriminative power in separating spam and non-spam tweets.

Authorized licensed use limited to: NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.



(a) Number of Lists (b) Number of Tweets (c) Number of Followings

(d) Number of Hashtags Per Tweet (e) Number of Characters Per Tweet (f) Number of Digits Per Tweet

Fig. 1: Cumulative Distribution Functions of Features

TABLE I: Extracted Features

Feature Name	Description
account_age	The age (days) of an account since its creation until the time of sending the most recent tweet
no_follower	The number of followers of this twitter user
no_following	The number of followings/friends of this twitter user
no_userfavourites	The number of favourites this twitter user received
no_lists	The number of lists this twitter user added
no_tweets	The number of tweets this twitter user sent
no_retweets	The number of retweets this tweet
no_hashtag	The number of hashtags included in this tweet
no_usermention	The number of user mentions included in this tweet
no_urls	The number of URLs included in this tweet
no_char	The number of characters in this tweet
no_digits	The number of digits in this tweet

The long time a spam tweet exists, the easier it can be exposure to victims. Thus, it is significant to detect spam tweets as early as possible. To mitigate the loss caused by spam, real-time detection is in demand. Consequently, we only focus on extracting light-weight features which can be used for timely detection. We can directly extract features from the collected tweets' data structure without or with little calculation, as all collected tweets were clearly structured by JSON format. We have totally extracted 12 features from our dataset as listed in TABLE I.

According to the place where the features were extracted, the 12 features can be divided into two categories, user-based features and tweet-based features. User-based features were extracted from the JSON object "user", including account_age, no_of follwers, no_of followings, no_userfavourites, no_lists, and no_tweets; while tweet-based features were extracted from the "tweet" object, including no_retweets, no_hashtags, no_usermentions, no_urls, no_chars, and no_digits.

D. Feature Statistics

We further investigated each feature's characteristics of differentiating spam and non-spam tweets. Fig 1 shows the Cumulative Distribution Function (CDF) of three user based features and three message based features.

We can see from Fig 1a that spammers are involved in more lists than normal users, so as to be exposed more to the public. Naturally, in order to spread more spam tweets, spammers send more tweets compared to non-spammers, as shown in Fig 1b. In terms of followings, Fig 1c shows that, spammers do like to follow more users than non-spammers. The aim is also to attract more attentions from victims to click their spam links.

As Fig 1d shows, non-spammers use less hashtags than spammers. There are about 80% non-spam tweets do not have hashtags embedded in their sent tweets, while the ratio in spam

Authorized licensed use limited to: NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.

tweets is only 60%. When it comes to the feature "Number of Characters Per Tweet", there is not much difference between spam tweets and non-spam tweets. The reason could be that spammers begin to imitate the posting behaviour of normal users. Fig 1f shows that spammers tend to use less digits than non-spammers. Due to the limit of pages, we only show six features' characteristics here. In general, the analysis of these features has showed us their discriminative power to detect Twitter spam.

IV. Spam Detection

In this section, we will evaluate the spam detection performance on our dataset by using six machine learning algorithms, *Random Forest, C4.5 Decision Tree, Bayes Network, Naive Bayes, k Nearest Neighbour,* and *Support Vector Machine.* We also sampled several different datasets for the experiments. The datasets are listed in TABLE II.

TABLE II: Sampled Datasets

Dataset	Sampling Method	NO. of Spam Tweets	NO. of Non-spam Tweets
I	Continuous	5000	5000
II	Continuous	5000	95000
III	Non-continuous	5000	5000
IV	Non-continuous	5000	95000

In TABLE II, we can see that the spam to non-spam ratio is 1:1 in Dataset I and III, while the ratio is 1:19 in Dataset II and IV. In previous works, most of the datasets are nearly evenly distributed, *i.e.* the spam to non-spam ration is nearly 1:1. However, there are around 5% spam tweets in Twitter [19]. The evenly distributed dataset cannot represent the Twittersphere. Consequently, we sampled Dataset II and IV which has a spam ration of 1:19 to simulate the real world scenario. These datasets can be divided into two groups based on the sampling method: Dataset I and II are randomly selected from the whole dataset, but the tweets were sent in a certain continuous time frame. On the other hand, the tweets in Dataset III and IV were not sent continuously. Instead, those tweets were totally independent from each other.

A. The Impact of Spam to Non-spam Ratio

In this section, We evaluate the impact of spam to non-spam ratio of the above-mentioned machine learning algorithms on Dataset I and II. Each classifier in this set of experiments was trained with a dataset of 1000 spam tweets and 1000 non-spam tweets. Then these trained classifiers were used to detect spam in the four sampled datasets. As in [7], we also used True Positive Rate (TPR), False Positive Rate (FPR) and F-measure to evaluate the performance of these classifiers.

As seen in TABLE III, most of the classifiers can achieve more than 90% TPR, expect Bayes Network and SVM, on both datasets. These classifiers can also reach satisfactory F-measure on Dataset I. However, the F-measures decrease dramatically when evaluating on Dataset II, *i.e.* when the spam to non-spam ration is 1:19.

To figure out why F-measure drops on Dataset II, TABLE IV outputs the confusion matrix of Random Forest when

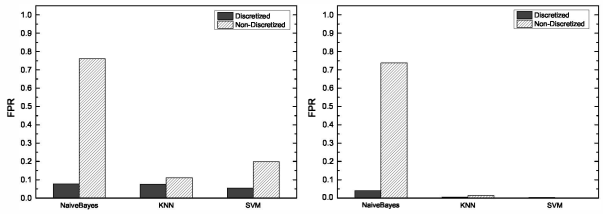
TABLE III: Performance Evaluation on Dataset I and II

Unit: %	Dataset I			Dataset II		
Classifier	TPR	FPR	F-measure	TPR	FPR	F-measure
RandomForest	92.9	5.6	93.6	92.9	7.1	56.6
C4.5	92.4	8.4	92	92.4	10.9	46.2
BayesNetwork	75.3	8.7	81.9	75.3	9.8	41.6
Naive Bayes	97.3	77.1	70.9	97.3	78.8	11.5
Knn	91.9	11.1	90.5	91.9	15.9	37.3
SVM	79.1	18.9	79.9	79.1	19.5	28.8

TABLE IV: Confusion Matrix of Random Forest on Both Datasets

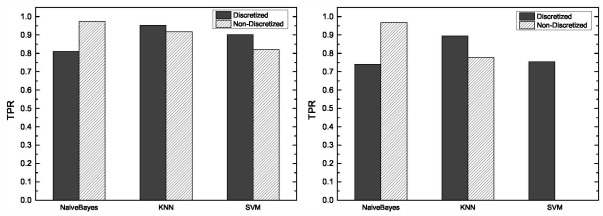
classified as –>	spam	non-spam		spam	non-spam
spam	4645	355		4645	355
non-spam	282	4718		6766	88234
	Dataset I			Dataset II	

evaluated on both datasets. Since the classifiers were trained by the same dataset, we can see that, there was no impact on the True Positives and False Negatives of spam class when the spam to non-spam ratio was changed, so Recall, which is define as the ratio of the number of tweets classified correctly as spam to the total number of real spam tweets, stayed the same. However, when more non-spam tweets were involved in the test, the number of False Positives increased exponentially. Thus, the precision, which is define as the ratio of the number of tweets classified correctly as spam to the total number of predicted spam tweets, decreased. As a result, F-measure, which is combination of precision and recall, decreased dramatically due the decrease of precision. Generally, we find that the F-measure of machine learning based classifiers is quite low as there are much more non-spam tweets than spam tweets.



(a) FPR on Dataset I (b) FPR on Dataset II

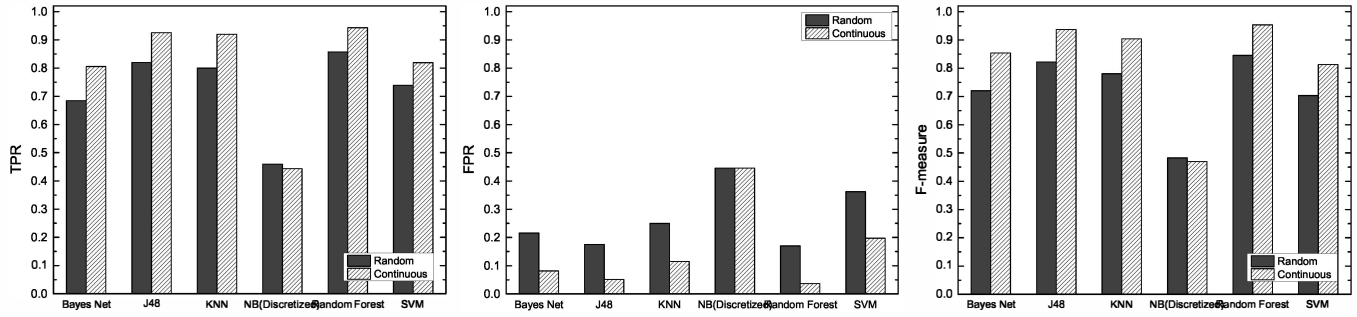
Fig. 2: False Negative Rate on Spam



(a) TPR on Dataset I (b) TPR on Dataset II

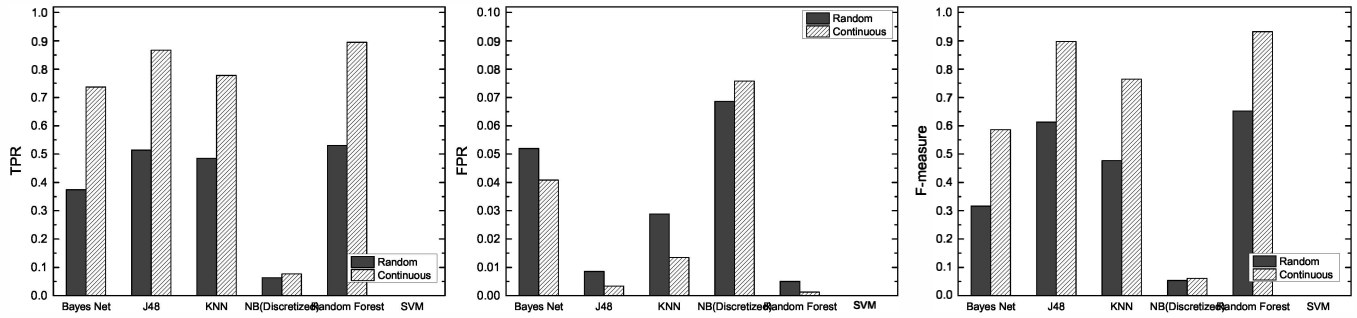
Fig. 3: True Positive Rate on Spam

Authorized licensed use limited to: NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.



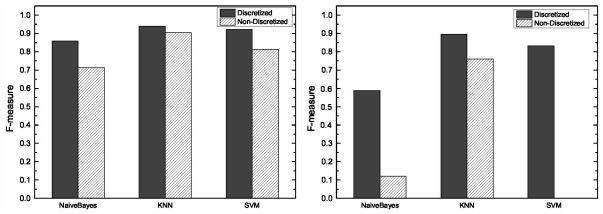
(a) TP Rate (b) FP Rate (c) F-measure

Fig. 5: Spam detection on Dataset I VS Dataset III



(a) TP Rate (b) FP Rate (c) F-measure

Fig. 6: Spam detection on Dataset II VS Dataset IV



(a) F-measure on Dataset I (b) F-measure on Dataset II

Fig. 4: F-measure on Spam

B. The Impact of Feature Discretisation

In this subsection, we evaluate the impact of feature discretisation of selected classifiers, such as Naive Bayes, kNN, and SVM, on discretised and non-discretised Dataset I and II. We can see from Fig 2 that the False Positive Rate of Naive Bayes decreases dramatically after discretisation, from 77% to 8% on Dataset I. Likewise, on Dataset II, the FPR declined from 75% to less than 5%. However, the performance of Naive Bayes also decreases in terms of True Positive Rate as shown in Fig 3. The TPR of Naive Bayes drops from 97.5% to 81% and from 96% to 74.5%, on Dataset I and II, respectively. When it comes to F-measure which is showed in Fig 4, the performance of Naive Bayes increases approximate 16% and 45% on Dataset I and II, respectively. Overall, feature discretisation has positive impact for Naive Bayes, especially when on Dataset II. Similarly, feature discretisation can help to improve performance for kNN and SVM on both datasets, especially for SVM, the F-measure increased from 0 to 80%, when applying discretisation on features. In general, feature discretisation can improve performance of classifiers for Twitter spam detection.

C. The Impact of Different Sampling Method

During our study, we find that classifiers' performance was better on the dataset which the tweets were sampled from a continuous period of time than that which the tweets were randomly selected. To further study this phenomenon, Dataset III and IV were sampled. We also perform 10-fold cross-validation on both datasets. The results are shown in Fig 5 and 6.

The results in Fig 5a indicate that the TP rates of all classifiers on Dataset I increased by approximate 10%, compared to the TPR on Dataset III, expect Naive Bayes. For example, the TP rates of C4.5 Decision Tree and kNN are 12% higher on Dataset I than those on Dataset III. In addition, most of these classifiers can reach 80% TP rate; some of them, such as C4.5 Decision Tree, kNN and Random Forest can even have over 90% TP rates when evaluated on Dataset I. Similarly, the FP rates on Dataset I drops significantly, especially for SVM, which drops from approximate 40% to less than 20%, with

Authorized licensed use limited to: NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.

a decrease of 20%. Most of the classifiers have a FP rate of less than 10%. In terms of F-measure, all classifiers evaluated on Dataset I, except Navie Bayes outperform those evaluated on Dataset III. Furthermore, several classifiers evaluated on Dataset I can achieve more than 90% F-measure, which is very effective in detecting Twitter spam.

Fig 6 shows the TP rates, FP rates and F-measures of all the classifiers evaluated on Dataset II and IV. The difference of TP rates on Dataset II and IV is significant, which is between 30% to 40%. When it comes to the metric of F-measure, the same difference exists. For instance, the F-measure of Random Forest evaluated on Dataset II can reach as high as 95%, which is 30% higher than it is on Dataset IV. In this set of experiments, we find that Naive Bayes and SVM work badly when on the datasets with 1:19 spam to non-spam ratio. Naive Bayes can only detect less than 10% spam tweets, while SVM miss all the spam tweets. We will put the problem why Naive Bayes and SVM cannot work well on imbalanced datasets as a future work.

In this subsection, we evaluate the performance of different classifiers on two kinds of datasets (random sampled and continuous sampled), and find that classifiers have much better performance in detection spam tweets on the continuous datasets. From this, we can see that there must be some kind of information correlation among the continuous tweets. We will further investigate the correlation in the near future.

V. Conclusion and Feature Work

In this paper, we firstly collected a large number of 600 million public tweets. Then we applied Trend Micro's Web Reputation System to label as many as 6.5 million spam tweets along with more than 6 million non-spam tweets. We also extracted light-weight features which are able to differentiate spam tweets and non-spam tweets from the labelled dataset. Furthermore, we used CDF figures to illustrate the characteristics of extracted features. We leveraged these features to machine learning based spam classification later in our experiments. To investigate the ability of spam detection of different classifiers, we sampled four different datasets to simulate various scenarios. Through our experiments, we found that classifiers' ability to detect Twitter spam would reduce when in a near real-world scenario, since the imbalanced data brings bias. In addition, classifiers' performance will be improved if features are discretised before classification. Furthermore, we note that classifiers can detect more spam tweets when the tweets were sent continuously rather than randomly selected tweets.

From our research in this paper, some problems in this field are also required to be further studied. Firstly, Naive Bayes and SVM worked much worse than other classifiers when on the datasets with 1:19 spam to non-spam tweets. Further research is demanded to investigate why these two classifiers cannot work well on our big datasets. Secondly, since classifier's ability to detect Twitter spam is better on the continuous datasets, there must exist correlation among the continuously sent tweets. This kind of information correlation is worth to be investigated as it can be made use to dramatically improve Twitter spam detection accuracy.

References

[1] H. Tsukayama, "Twitter turns 7: Users send over 400 million tweets per day," *Washingon Post*, March 2013. [Online]. Available: http://articles.washingtonpost.com/2013-03-21/business/37889387_1_tweets-jack-dorsey-twitter

[2] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammer on twitter," in *Seventh Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, July 2010.

[3] D. Goodin, "Mystery attack drops avalanche of malicious messages on twitter," *Ars technica*, Aprial 2014. [Online]. Available: http://arstechnica.com/security/2014/04/mystery-attack-drops-avalanche-of-malicious-messages-on-twitter/

[4] K. Thomas, C. Grier, D. Song, and V. Paxson, "Suspended accounts in retrospect: an analysis of twitter spam," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '11. New York, NY, USA: ACM, 2011, pp. 243–258.

[5] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, "Design and evaluation of a real-time url spam filtering service," in *Proceedings of the 2011 IEEE Symposium on Security and Privacy*, ser. SP '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 447–462.

[6] E. Tan, L. Guo, X. Zhang, and Y. Zhao, "Unik: Unsupervised social network spam detection," in *Proceedings of 22nd ACM International Conference on Information and Knowledge Management*, San Fransisco, USA, October 2013.

[7] C. Yang, R. Harkreader, and G. Gu, "Empirical evaluation and new design for fighting evolving twitter spammers," *Information Forensics and Security, IEEE Transactions on*, vol. 8, no. 8, pp. 1280–1293, 2013.

[8] S. Lee and J. Kim, "Warningbird: A near real-time detection system for suspicious urls in twitter stream," *IEEE Transactions on Dependable and Secure Computing*, vol. 10, no. 3, pp. 183–195, 2013.

[9] C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu, "Analyzing spammers' social networks for fun and profit: a case study of cyber criminal ecosystem on twitter," in *Proceedings of the 21st international conference on World Wide Web*, ser. WWW '12. New York, NY, USA: ACM, 2012, pp. 71–80.

[10] J. Song, S. Lee, and J. Kim, "Spam filtering in twitter using sender-receiver relationship," in *Proceedings of the 14th international conference on Recent Advances in Intrusion Detection*, ser. RAID'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 301–317.

[11] M. Egele, G. Stringhini, C. Kruegel, and G. Vigna, "Compa: Detecting compromised accounts on social networks," in *Annual Network and Distributed System Security Symposium*, 2013.

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

[13] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC '10. New York, NY, USA: ACM, 2010, pp. 1–9.

[14] S. Yardi, D. Romero, G. Schoenebeck, and D. Boyd, "Detecting spam in a twitter network," *First Monday*, vol. 15, no. 1-4, January 2010.

[15] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 591–600.

[16] X. Zhang, S. Zhu, and W. Liang, "Detecting spam and promoting campaigns in the twitter social network," in *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, 2012, pp. 1194–1199.

[17] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th international conference on World wide web*, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 675–684.

[18] J. Oliver, P. Pajares, C. Ke, C. Chen, and Y. Xiang, "An in-depth analysis of abuse on twitter," Trend Micro, 225 E. John Carpenter Freeway, Suite 1500 Irving, Texas 75062 U.S.A., Tech. Rep., September 2014.

[19] C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@spam: the underground on 140 characters or less," in *Proceedings of the 17th ACM conference on Computer and communications security*, ser. CCS '10. New York, NY, USA: ACM, 2010, pp. 27–37.

Authorized licensed use limited to: NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL. Downloaded on January 05,2022 at 16:18:37 UTC from IEEE Xplore. Restrictions apply.