Impact of Tweet Features and Machine-Learning Classifiers for Twitter Spam Detection

S. Nitheesh Prabu , Abhishek Pal, S. Sundar Ram, S. Karthika

Abstract: Twitter changed how people get their everyday news and has provided a different platform of communication for everyone. It is this capacity of Twitter that attracts a lot of spammers to the platform. Twitter has an anti-spamming team and also encourages its users to report tweets, which they feel are spam. Even though this helps in identifying spam, it does not guarantee real time protection of the user. A number of mechanisms have been proposed to block spam to keep Twitter a safer place. Recent studies have directed efforts on detecting Twitter spam by applying machine learning algorithms. This paper is a study of such mechanisms. The dataset consists of 1,00,000 tweets which had been extracted from a tweet dataset containing 600 million tweets, out of which 6.5 million were spam. Each tweet was described using 12 features. The problem was then converted into a binary classification problem in the feature space. Then the importance of the features was analysed, studying the results of multiple classifiers and metrics.

Keywords: AdaBoost; Classification; Decision Tree; Logistic Regression; Machine Learning; Naïve Bayes; Random Forest; Social networking; SVM; Twitter Spam.

I. INTRODUCTION

Social networking sites (e.g., Twitter and Facebook) are popular tools for online communication among people. Internet users prefer these sites for communication and interaction. Unfortunately, this also attracts spammers who stream waves after waves of unwanted tweets that expose the users to harmful websites or services and other malicious behaviour (e.g., tweets containing scam, phishing, and other malicious URLs), leading to great inconvenience. Social spam is unwanted content that is posted on social media, or any website with any form of content created by users (posts, status updates, comments, chat, etc.). It can appear in many forms, including bulk unwanted messages, vulgarity, abuses, malicious links etc. The impact of spam is significant. Typically, a spam message cannot be identified just by looking at the URL because of the URL shorten services used. These spam messages are seen by all followers and friends of the account in question, and causes inconvenience and misdirection in public and trending topics. For this reason, trending topics are largely abused by spammers in order to gain more traction on their malicious behaviour, directing users to irrelevant and unnecessary websites.

Revised Manuscript Received on March 25, 2019.

S. Nitheesh Prabu, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, TN 603110

Abhishek Pal, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, TN 603110

S. Sundar Ram, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, TN 603110

S. Karthika, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, TN 603110Twitter spam is not a new phenomenon. It is the popularity of the growing amount of social networking sites that has attracted a large number of spammers. The current amount of spam occurs even in light of the spam handling efforts taken by Twitter. Therefore, it becomes increasing imperative to devise more efficient and effective spam handling mechanisms in order to tackle this inadvertently growing problem. Consequently, the research community and Twitter itself have proposed few schemes to reduce spam. For instance, Twitter has certain rules that automatically suspend accounts that behave automatically. Users are encouraged to mark spam tweets as they come across them. Although in the long run this may seem like a fit, the number of tweets that arrive at the website is still huge. Therefore, it becomes a cumbersome and time-consuming process to manually detect and mark each and every tweet as spam and not spam. This does not work well in real time also. Alternatively, classifying a tweet as spam or not spam, based on certain features to the tweet stream and history of the tweet, may prove to be the answer for this problem.

II. RELATED WORK

The growing problem of Twitter spam has already demanded attention from many researchers. After studying the characteristics of spam, researchers have proposed significant works in order to identify Twitter spam.

A. Detecting Twitter Spam

Lots of works have been introduced to detect Twitter spam. A majority of these implement machine learning algorithms to distinguish spam and non-spam, including [4], [1], [5], [9], and [8]. Many of these research works have used user based and tweet-based features, like number of followers, URLs, age etc. The authors in [4] have focused on combining NLP, machine learning and URL analysis for Twitter spam classification. This combination provides superior results and improved accuracy compared to individual application of the mentioned techniques. Spam Detection in [7] is done using traditional classifiers like Naïve Bayes, SMO, KNN and Random Forest. Among the classifiers used, Random Forest has given the best results. These features were more distinctive than the features in similar research. The downside is that collecting these features is an extremely cumbersome task. Instead, [9] and [6] relied on URL based features in tweets and some features from the landing page to detect spam, along with domain information.

In [6], characteristics of URL redirect chains were studied, as they showed greater discriminative power when used to classify spam. However, these techniques only identify spam that contains URLs in the form of hyperlinks [3]. Messages that contain just text or hardcoded URLs will be missed by this system. A model-based spam detection scheme was thus proposed by [3]. Spam detection via sentiment analysis was done in [8]. Learning based and lexicon-based techniques for sentiment analysis were discussed. While spam messages can be posted without URLs, the majority of them on Twitter contain URLs [3]. Therefore, this research paper is restricted to analysing tweets that contain URLs.

III. METHODOLOGY

A dataset as ground-truth (annotated instances with class labels for referencing) is needed to perform a number of challenging machine learning-based streaming spam tweets detection tasks. A problem as big as detection of Twitter spam does require a really big dataset. The dataset used contains 1,00,000 tweets and 12 features. The authors obtained the dataset used in [2]. The analysis was performed based on this as the ground truth. This research work examines the importance of features by applying traditional Machine Learning algorithms like Decision Trees, Random Forest, Support Vector Machine (SVM) and Logistic Regression.

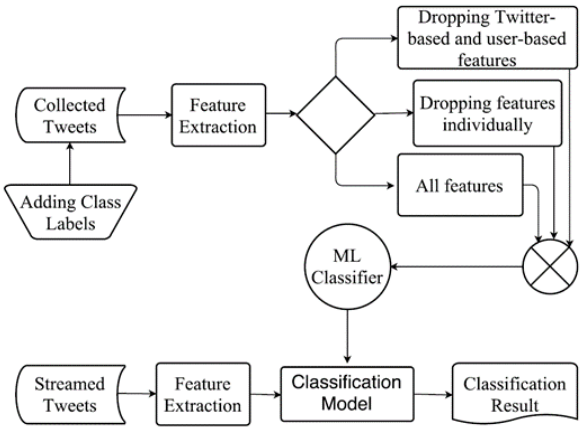

Fig. 1: ML based spam detection process.

A. Ground Truth

The dataset was obtained from the authors of [2]. They used the services of Trend Micro's Web Reputation Service to ascertain which URLs were malicious. The authors kept this as their ground truth. Trend Micro WRS obtain their data through their opt-in URL filtering records. WRS collects the URLs in real time and analyses them. WRS uses many frontier technologies and even visits the links manually if needed for the purpose of analysing and labelling. This leaves us with enough support to claim this as our ground truth. Over 600 million URLs were collected by the authors of [2]. They were given to WRS for the purpose of labelling. WRS labelled 6.5 million as spam. The authors of [2] provided us with this dataset which consisted of more than 1,00,000 tweets and each tweet was described using 12 features.

B. About the Dataset

The dataset contained 1,00,000 tweets out of which 95,000 were non-spam and 5,000 were spam. Each tweet was marked as spam/non-spam and had 12 other features as shown in Table 1. According to the object where the features were extracted, the 12 features can be divided into two categories, user-based features, and tweet-based features [2]. User-based features consisted of features such as account_age (the age of an account since its creation until the time of sending the most recent tweet), no_of_followers (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), and no_tweets (the number of tweets this twitter user sent). Tweet-based features includes no_retweets (the number of retweets this tweet), no_hashtags (the number of hashtags included in this tweet), no_usermentions (the number of user mentions included in this tweet), no_urls (the number of URLs included in this tweet), no_chars (the number of characters in this tweet), and no_digits (the number of digits in this tweet).

Table 1: Features in the dataset

Feature Name	Description
account age	Age of account since its creation
no_follower	Number of users following this user
no_following	Number of users followed by the user
no_userfavourites	Number of favorites of the user
no_lists	Number of lists this user was added
no_tweets	Total number of tweets by the user
no_retweets	Total number of retweets by the user
no_hashtag	Total number of hashtags in the tweet
no_usermention	Number of users mentioned in the tweet
no_urls	Number of URLs hardcoded in the tweet
no_char	Number of characters in the tweet
no_digits	Number of digits in the tweet

C. Algorithms and Metrics Used

The dataset was labelled, as spam/non-spam and contain 12 features. Since the dataset had only 2 classes, the problem was converted to a binary classification problem. Various Machine Learning algorithms such as Random Forest, Logistic Regression, Decision Tree, AdaBoost, Naïve Bayes and SVM were used. Many metrics were computed and the results were compared. The algorithms were tested on the dataset using stratified k-fold cross validation. The advantage of this method is that all data are used for both training and validation, and each data is used for validation exactly once.

The authors computed a total of 5 metrics. The metrics computed were Classification accuracy, Standard Deviation, F1-measure (and in turn, Precision and Recall), Logarithmic Loss and Area under ROC curve.

IV. RESULTS OBSERVED

The importance of the features was examined. This was done in 3 ways. Firstly, a single feature was dropped to check if that feature was important in the classification of spam. Next, the features were split into 2 groups, user-based and tweet-based features. The importance of the features was tested by testing the features on various Machine Learning algorithms. Also, different performance metrics were evaluated and the results were compared.

A. Dropping features individually

A feature was dropped and the test was carried out on the remaining 11 features. This was repeated on all the 12 features. Various ML algorithms were applied and results between these algorithms were computed as in Table 2.

Table 2: Accuracy for various classifiers when features were dropped individually (per cent)

Feature Dropped	Decision Tree	LogRegression	Random Forest	Ada Boost	Naïve Bayes	SVM
account_age	94.99	94.92	96.90	94.94	94.88	67.28
no_follower	94.64	95.05	96.69	94.86	94.55	84.12
no_following	94.90	95.03	96.84	94.89	94.84	86.10
no_userfavourites	94.90	95.07	96.93	94.93	94.84	81.49
no_lists	95.08	95.03	96.95	95.02	94.99	71.25
no_tweets	94.94	95.02	96.83	94.94	94.94	78.73
no_retweets	94.79	95.04	96.95	94.91	95.01	71.67
no_hashtag	94.39	95.09	96.79	94.86	94.91	88.86
no_usermention	95.12	95.09	96.93	95.04	95.05	68.33
no_urls	95.25	95.09	96.96	95.09	95.10	79.24
no_char	95.14	95.09	96.92	94.92	95.07	79.01
no_digits	95.07	95.09	96.90	95.13	95.07	87.83

It can be observed that the classification accuracy is almost identical for all features and all classifiers. This signifies that no single feature plays an important role in classifying the tweets.

B. Dropping tweet-based features versus user-features

Here, the importance of the two groups was tested. Various ML algorithms were simulated on these 2 groups and the results were observed. From Table 3, it can be observed that tweet-based features fair marginally better than user-based features in classifying the tweets as spam and not-spam. This can be attributed to fact that tweet-based features contain more attributes that are related to tweets that contain spam, like the number of URLs in a tweet, the number of digits, number of characters, etc. But this difference is not profound enough to favour one group over the other in classification.

Table 3: Accuracy when user-based and tweet-based features were dropped alternatively

Feature Dropped	Decision Tree	Logistic Regression	Random Forest	Ada Boost	Naïve Bayes	SVM
User-Based	95.02	94.92	96.58	94.94	94.53	77.74
Tweet-Based	95.11	95.01	96.69	94.86	94.55	77.89

C. Combining tweet-based features and user-features

It was observed that the highest and best possible results were obtained when both the tweet-based and user-based features were combined to classify spam. The classifiers together produced an accuracy of 92.56%, which is a very acceptable rate. Random Forest provided the best accuracy out of all the other classifiers, across a 10-fold cross validation. From Table 4, it can be observed that the standard deviation for all classifiers is pretty low. This indicates that the distribution of ranges in the two classes of the dataset is very low. The consequences of this might be that the model may overfit to one class and fail to classify the second class totally. From Table 4, the Log Loss of the Decision Tree classifier is very less compared to those of the other classifiers. This indicates that the number of false classifications is very less in the Decision Tree as compared to other classifiers, and the overall accuracy in terms of misclassification is better. It is also observed that the accuracy of Random Forest classifier is slightly greater than that of Decision Tree.

Classifier	Accuracy	Standard Deviation	Log Loss	Area Under the Curve
Decision Tree	95.29%	0.00188	-1.624	0.7654
Logistic Regression	95.01%	0.0024	-0.208	0.6487
Random Forest	96.92%	0.0017	-0.3248	0.8914
Ada Boost	95.15%	0.0023	-0.6574	0.8438
Naïve Bayes	95.03%	0.0023	-0.1865	0.7228
SVM	78.0%	0.118	-0.102	0.527

This is because the accuracy measure does not take into account the misclassifications (False Positives and False Negatives). This goes to show that even though the accuracy is high, it might not necessarily mean that the classifier will predict most of the samples correctly.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

One reason why this might have happened is due to the lesser number of samples in the spam class on both the training and test datasets. The overall accuracy will not alter much, even if all the instances of spam class were misclassified. On the other hand, the Log Loss will vary dramatically. The area under the curve metric again indicates that the Random Forest is the best classifier out of the others. From Table 4, SVM seems to have not been able to fit to the data, and is randomly guessing the classes, because it has a RUC value of 0.527.

A number of other metrics were also used to grade the performance of the algorithms. Table 5 lists the F-Score, Precision and Recall of the various classifiers used.

Table 4: F-Score, Precision and Recall for the classifiers

Classifier	F-Score		Precision		Recall	
	Non-Spam	Spam	Non-Spam	Spam	Non-Spam	Spam
Decision Tree	97.0	55.0	97.0	55.0	97.0	55.0
Logistic Regression	97.0	4.0	95.0	35.0	100.0	2.0
Random Forest	98.0	61.0	97.0	92.0	100.0	46.0
Ada Boost	97.0	15.0	95.0	61.0	100.0	9.0
Naïve Bayes	97.0	0.0	95.0	0.0	100.0	0.0
SVM	77.0	7.0	94.0	4.0	66.0	27.0

The dataset used is highly skewed, with only about 1% of its entirety being labelled spam. But both classes are equally important. In such a situation, the classifier with a higher F-Score on both the classes will prove to be the more effective one. It can be observed that the Random Forest classifier has the best possible combination of F-Score for both the classes. A close second is the Decision Tree. This is intuitive, as the Random Forest is in fact a multitude of decision trees. Other classifiers like AdaBoost, SVM and Logistic Regression have poor F-Score on the non-spam class (skewed dataset). The Naïve Bayes classifier has classified the entirety of test data as non-spam. Naïve Bayes in general requires an equal distribution among the classes while training in order to obtain maximal results.Although it is important to classify spam, it is equally important to not misclassify non-spam as spam, and the precision and recall metrics allow us to verify if the classifiers do that. Precision indicates the probability that a positive prediction is correct, and recall indicates how likely the classifier is to be able to predict a positive class as positive. It can be observed that the Decision Tree and Random Forest classifiers provide better overall precision and recall compared to other classifiers. Once again, classifiers like Naïve Bayes and Logistic Regression have very poor recall on the spam class, whereas on the non-spam class, the recall is very good.

V. CONCLUSION AND FUTURE WORK

This paper focuses on the evaluation of the various machine learning classifiers in detecting spam tweets. This was done by comparing the accuracy and various measures like F1-Score, Precision and Recall for the various algorithms. Out of all the classifiers, Random Forest seemed to have a slight edge in terms of accuracy. Some classifiers, like Naïve Bayes, which is actually predominantly used in text classification, failed to detect spam. In hindsight, this effect is likely due to the nature of the dataset - which is quantitative (numeric). As a result, the learning was highly dependent on the distribution of classes in the dataset. Adding to the above point, it is noted that the dataset is indeed quite skewed towards the non-spam class. The various features that were extracted from the tweets were also tested for their discriminative power. This was done in three ways - dropping each feature one by one, dropping user-based and tweet-based features alternatively, and using all the features. The authors inferred that out of the 12 selected features, no particular feature was better than the other in terms of discriminative power per se. The overall accuracy did not alter a significant amount, but it did not reduce either. Thus, it can be said that all the selected features are very good. Perhaps this might change if the distribution of the dataset changes too, as there might be more diversity in the features. Owing to the above-mentioned points, the distribution of spam to non-spam tweets might need to be improved so as to build a better, accurate model, with good precision and recall. The authors will work on a more diversely distributed dataset in the future.

Acknowledgement

The authors would like to thank the authors of [2] for providing us with their well curated dataset for the further research on tweet features.

REFERENCES

1. Benevenuto F, Magno G, Rodrigues T & Almeida V, "Detecting Spammers on Twitter", *CEAS 2010 - Seventh annual Collaboration, Electronic messaging, AntiAbuse and Spam Conference*, (2010).
2. Chen C, Zhang J, Chen X, Xiang Y & Zhou W, "6 Million spam tweets: A large ground truth for timely Twitter spam detection", *IEEE International Conference on Communications (ICC)*, (2015), pp:7065-7070.
3. Egele M, Stringhini G, Kruegel C & Vigna G, "Detecting compromised accounts on social networks", *20th Annual Network & Distributed System Security Symposium (NDSS)*, (2013).
4. Kandasamy K & Koroth P, "An integrated approach to spam classification on Twitter using URL analysis, natural language processing and machine learning techniques", *IEEE Students' Conference on Electrical, Electronics and Computer Science*, (2014).
5. Lee K, Caverlee J & Webb S, "Uncovering social spammers: social honeypots + machine learning", *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, (2010), pp:435-442
6. Lee S & Kim J, "A near real-time detection system for suspicious URLs in Twitter stream", *IEEE Transactions on Dependable and Secure Computing*, Vol. 10, (2013), pp:183-195.

7. McCord M & Chuah M, "Spam detection on twitter using traditional classifiers", *International Conference on Autonomic and Trusted Computing (ATC 2011)*, (2011), pp:175-186.
8. Bhuta S, Doshi A, Doshi U & Narvekar M, "A Review of Techniques for Sentiment Analysis of Twitter Data", *International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, (2014), pp:583-591.
9. Stringhini G, Kruegel C & Vigna G, "Detecting spammers on social networks", *Proceedings of the 26th Annual Computer Security Applications Conference*, (2010), pp:1-9.
10. Rajesh, M., and J. M. Gnanasekar. "Path Observation Based Physical Routing Protocol for Wireless Ad Hoc Networks." Wireless Personal Communications 97.1 (2017): 1267-1289.
11. Rajesh, M., and J. M. Gnanasekar. "Sector Routing Protocol (SRP) in Ad-hoc Networks." Control Network and Complex Systems 5.7 (2015): 1-4.
12. Rajesh, M. "A Review on Excellence Analysis of Relationship Spur Advance in Wireless Ad Hoc Networks." International Journal of Pure and Applied Mathematics 118.9 (2018): 407-412.
13. Rajesh, M., et al. "SENSITIVE DATA SECURITY IN CLOUD COMPUTING AID OF DIFFERENT ENCRYPTION TECHNIQUES." Journal of Advanced Research in Dynamical and Control Systems 18.
14. Rajesh, M. "A signature based information security system for vitality proficient information accumulation in wireless sensor systems." International Journal of Pure and Applied Mathematics 118.9 (2018): 367-387.
15. Rajesh, M., K. Balasubramaniaswamy, and S. Aravindh. "MEBCK from Web using NLP Techniques." Computer Engineering and Intelligent Systems 6.8: 24-26.