Spam Detection on Twitter Using Traditional Classifiers

M. McCord
CSE Dept
Lehigh University
19 Memorial Drive West
Bethlehem, PA 18015, USA
mpm308@lehigh.edu

M. Chuah
CSE Dept
Lehigh University
19 Memorial Drive West
Bethlehem, PA 18015, USA
chuah@cse.lehigh.edu

ABSTRACT
Social networking sites have become very popular in recent years. Users use them to find new friends, updates their existing friends with their latest thoughts and activities. Among these sites, Twitter is the fastest growing site. Its popularity also attracts many spammers to infiltrate legitimate users' accounts with a large amount of spam messages. In this paper, we discuss some user-based and content-based features that are different between spammers and legitimate users. Then, we use these features to facilitate spam detection. Using the API methods provided by Twitter, we crawled active Twitter users, their followers/following information and their most recent 100 tweets. Then, we analyzed the collected dataset and evaluated our detection scheme based on the suggested user and content-based features. Our results show that among the four classifiers we evaluated, the Random Forest classifier produces the best results. Our results based on the 100 most recent tweets also show that spam detection based on our suggested features can achieve 95.7% precision and 95.7% F-measure using the Random Forest classifier.

Categories and Subject Descriptors
J. 4[Computer Applications]: Social and behavioral sciences.

General Terms
Algorithms, Experimentation, Security.

Keywords
Social network security, spam detection, classifier, machine learning.

1. INTRODUCTION
Online social networking sites such as Facebook, LinkedIn and Twitter allow millions of users to meet new people, stay in touch with friends, establish professional connections and more. According to the report in [9], Twitter is the fastest growing social networking site among all the social networking sites. Twitter provides a micro-blogging service to users where users can post their messages, called tweets. Each tweet is limited to 140 characters and only text and HTTP links can be included in the tweets. Such tweet exchanges allow friends/colleagues to communicate and stay connected.

Twitter users have different levels of awareness with respect to security threats hidden in social networking sites. For example, a previous study has showed that 45% of users on a social networking site readily click on links posted by any friend in their friendlists' accounts, even though they may not know that person in real life [11]. Thus, spammers are attracted to use Twitter as a tool to send unsolicited messages to legitimate users, post malicious links, and hijack trending topics. Spam is becoming an increasing problem on Twitter as well as on other online social networking sites. A study shows that more than 3% of the messages are spam on Twitter [1,2,15]. Even the trending topics, which are the most tweeted-about-topics on Twitter, were attacked by spammers. A trending-topic attack reported in [3] forced Twitter to temporarily disable the trending topics so as to remove the offensive terms.

To deal with increasing threats from spammers, Twitter provides several ways for users to report spam. A user can report a spam by clicking on the "report as spam" link in their home page on Twitter. The reports are investigated by Twitter and the accounts being reported will be suspended if they are found to be spam. Another publicly available method is to post a tweet in the "@spam @username" format where @username mentions a spam account. However, even this service is also abused by spammers. Some Twitter applications also allow users to flag possible spammers. Additional methods and applications to reduce Twitter spam are described in [4]. Twitter also puts efforts into closing suspicious accounts, and filtering out malicious tweets. However, some legitimate Twitter users complain that their accounts were mistakenly suspended by Twitter's cleaning efforts [5]. All these ad hoc methods depend on users to identify spam manually based on their own experience. We need some tools that can automatically identify spammers. In addition, we need more accurate but efficient spam detection methods to avoid causing inconvenience to legitimate users.

ATC'11, Sept 2-4, 2011, Banff, Canada.

In this paper, we first study the differences between the tweets published by spammers and legitimate users. Our goal is to identify useful features that can be used in traditional machine learning schemes to automatically distinguish between spamming and legitimate accounts. The major contributions of this paper are as follows:

- We propose using user-based features and content-based features to facilitate spam detection
- We compare the performance of four traditional classifiers, namely Random Forest, Support Vector Machine, Naïve Bayesian and K-Nearest Neighbor classifiers, in their abilities to distinguish suspicious users from normal ones.
- We developed a prototype to evaluate the detection scheme based on our suggested features. The results show that our spam detection system has a 95.7% precision and 95.7% F-measure using the Random Forest Classifier.

The rest of the paper is organized as follows. In Section 2, we give some background about the Twitter site, and discuss related work. In Section 3, we discuss the various user-based and content-based features we proposed. In Section 4, we describe the characteristics of these user-based and content-based features based on the dataset we have collected. In Section 5, we first describe how our spam detection method works. Then, we report our evaluation results. We conclude in Section 6.

2. Background And Related Work

2.1 The Twitter Social Network

Twitter is a social networking site just like Facebook and MySpace except that it only provides a microblogging service where users can send short messages (referred to as tweets) that appear on their friends' pages. A Twitter user is only identified by a username and optionally by a real name. A Twitter user can start "following" another user X. Consequently, that user receives user X's tweets on her own page. User X who is "followed" can follow back if she so desires. Tweets can be grouped using hashtags which are popular words, beginning with a "#" character. Hashtags allow users to efficiently search tweets based on topics of interest. When a user likes someone's tweet, she can "retweet" that message. As a result, that message is shown to all her followers. A user can decide to protect her profile. By doing so, any user who wants to follow that private user needs her permission. Twitter is the fastest growing social networking site with a reported growth rate of 660% in 2009 [9].

2.2 Related Work

Since social networks are strongly based on the notion of a network of trust, the exploitation of this trust might lead to significant consequences. In 2008, an experiment showed that 41% of the Facebook users who were contacted acknowledged a friend request from a random person [10]. L. Bilge et al [11] show that after an attacker has entered the network of trust of a victim, the victim will likely click on any link contained in the messages posted, irrespective of whether she knows the attacker in real life or not. Another interesting finding by researchers [12] is that phishing attempts are more likely to succeed if the attacker uses stolen information from victims' friends in social networks to craft their phishing emails. For example, phishing emails from shoppybag were often sent from a user's friendlist and hence a user is often tricked into believing that such emails come from trusted friends and hence willingly provides login information of his/her personal email account. In [13], the authors created a popular hashtag on Twitter and observed how spammers started to use it in their messages. They discuss some features that might distinguish spammers from legitimate users e.g. node degree and frequency of messages. However, merely using simple features like node degree and frequency of messages may not be enough since there are some young Twitter users or TV anchors that post many messages.

A larger spam study was reported in [14]. The authors in [14] generate honey profiles to lure spammers into interacting with them. They create 300 profiles each on popular social networking sites like Facebook, Twitter and MySpace. Their 900 honey profiles attract 4250 friends request (mostly on Facebook) but 361 out of 397 friend requests on Twitter were from spammers. They later suggested using features like the percentage of tweets with URLs, message similarity, total messages sent, number of friends for spam detection. Their detection scheme based on the Random Forest classifier can produce a false positive rate of 2.5% and a false negative rate of 3% on their Twitter dataset.

In [15], the authors propose using graph-based and content-based features to detect spammers. The graph-based features they use include the number of followers, the number of friends (the number of people you are following) and a reputation score which is defined as the ratio between the number of followers over the total sum of the number of followers and the number of people a user is following. The conjecture is that if the number of followers is small compared to the amount of people you are following, the reputation is small and hence the probability is high that the associated account is spam. The content-based features they use include (a) content similarity, (b) number of tweets that contain HTTP links in the most recent 20 tweets, (c) the number of tweets that contain the "@" symbols in a user's 20 most recent tweets, (d) the number of tweets that contain the "#" hashtag symbol. Using a Bayesian classifier, the author found that out of the 392 users that are classified as

spammers, 348 are really spam accounts and 44 users are false positives so the precision of his spam detection scheme is 89%.

3. User-Based & Content-Based Features

In this section, we discuss the features we extract from each Twitter user account for the purpose of spam detection. The features extracted can be categorized into (i) user-based features and content-based features. User-based features are based on a user's relationships e.g. those whom a user follow (referred to as friends), and those who follow a user (referred to as followers) or user behaviors e.g. the time periods and the frequencies when a user tweets.

3.1 User-Based Features

In Twitter, you can build your own social network by following friends and allowing others to follow you. Spam accounts try to follow large amount of users to gain their attention. The Twitter's spam and abuse policy [6] says that, "if you have a small number of followers compared to the amount of people you are following", then it may be considered as a spam account. Three user-based features, namely the number of friends, the number of followers, and the reputation of a user are computed for spam detection in [15]. The reputation of a user is defined in [15] as

$$R(j) = \frac{n_i(j)}{n_i(j) + n_O(j)} \quad (1)$$

where $n_i(j)$ represents the number of followers user j has and $n_o(j)$ represents the number of friends ("following") user j has. However, in our work, we only use the number of followers and the number of "following" as part of our user-based features.

3.1.1 Distribution of Tweets over 24-hour period

In addition, we define statistics that are based on the percentage distribution of tweets in each of the 8 3-hour periods within a day (e.g. 1st time slot is from 0-3hr, 2nd is from 3-6 hr, etc) posted by a user. Our conjecture is that spammers tend to be most active during the early morning hours while regular users will tweet much less during typical sleeping hours. We compute these 8 statistics based on the local time associated with the location reported in a user's profile.

3.2 Content-Based Features

For content-based features, we use some obvious features e.g. the average length of a tweet. Additional content-based features are described in subsequent subsections.

3.2.1 Number of URLs

Since Twitter only allows a message with a maximum length of 140 characters, many URLs included in tweets are shortened URLS. Spammers often include shortened URLs in their tweets to entice legitimate users to access them. Twitter filters out the URLs linked to known malicious sites. However, shortened URLs can hide the source URLs and obscure the malicious sites behind them. While Twitter does not check these shorten URLs for malware, any user's updates that consist mainly of links are considered spam according to Twitter's policy. In [15], the authors use the percentage of tweets containing HTTP links in the user's 20 most recent tweets. If a tweet contains the sequence of characters "http;// or www., this tweet is considered containing a HTTP link. In our work, we use the number of HTTP links that are contained in a user's 100 most recent tweets.

3.2.2 Replies/Mentions

A user is identified by a unique username and can be referred to using the @username format in tweets on Twitter. Each user can send a reply message to another user using the @username+message format where @username is the message receiver. Each user can reply to anyone on Twitter whether they are his friends/followers or not. He can also mention another @username anywhere in his tweet, rather than just at the beginning. Twitter automatically collects all tweets containing a username in the @username format in his replies tab. The reply and mention features are designed to help users track conversation and discover each other on Twitter.

However, spammers often abuse this feature by including many @usernames as unsolicited replies or mentions in their tweets. If a user includes too many replies/mentions in his tweets, Twitter will consider that account as suspicious. The number of replies and mentions in a user account is measured by the number of tweets containing the @symbol in the user's 20 most recent tweets in [15]. However, we used a feature that measures the total number of replies/mentions in the most 100 recent tweets for each user.

3.2.3 Keywords/Wordweight

Since we observe that the contents in spammers' tweets contain similar words, we define two metrics to help identify spammers. First, we created a list of spam words that are often found in spammers' tweets and the associated probabilities of these words, and a list of popular words in legitimate tweets and the associated probabilities of these words. Our two defined metrics using this information are: (a) the keywords metric which counts the average number of spam words found in the 100 most recent tweets. For example, if we find a total of 50 spam words in the 100 most recent tweets, the keyword metric of that user will be 50/100, (b) the word weight metric which is defined as the difference between the sum of weighted probabilities of spam words and the sum of weighted probabilities of legitimate words found in a user's tweets. Assume that the word "hello" appears in a user's tweet and the weight of the word "hello" in the spamword list is 0.2 while the weight of that same word "hello" in the regular word list is 0.1, then the wordweight based on this word "hello" will be 0.2-0.1=0.1. The final wordweight is the sum of the weights for

all words from the spamword and regular word lists that can be found in a user's tweets.

3.2.4 *Retweets/Tweetlen*

Twitter allows users to retweet tweets generated by other users. All retweets start with the symbol @RT. The number of retweets in the 20-100 most recent tweets of a user is also used as one of the content-based features in our spam detection system. The average tweet length is also used as a content feature.

3.2.5 *Hashtags*

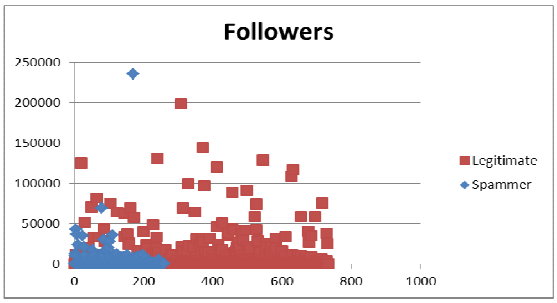
Trending topics are the most-mentioned terms on Twitter at that moment, this week or this month. Users can use the hashtag, which is the #symbol followed by a term describing or naming the topics, to a tweet. If there are many tweets containing the same term, the term will become a trending topic. Spammers often post many unrelated tweets that contain the trending topics to lure legitimate users to read their tweets. Twitter considers an account as spam "if a user posts multiple unrelated updates to a topic using the # symbol". The number of tweets which contains the symbol "#" in a user's 100 most recent tweets is used as one of the content-based features in [15]. However, in our work, we count the total number of hashtags in the 100 most recent tweets of each user.

4. Analysis of Collected Data

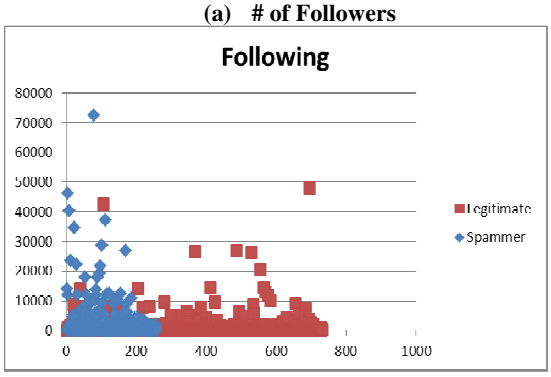
To evaluate the detection method, we randomly pick about 1000 Twitter user accounts and manually label them to two classes: spam and non-spam. Each user account is manually evaluated by reading the 20, 50, 100 most recent tweets posted by the user and checking the number of followers and following in his/her user profile page. Then, we extracted all the relevant user-based and content-based features that we have described in Section 3. Since we observe that we get better classification results with the most 100 tweets, we only report the results we get with the most 100 tweets. Fig 1(a) to (1d) show the characteristics of the user-based features, namely (a) the number of followers, (b) the number of "following" (or friends as defined in [15]), (c) the reputation, and (d) average posting percentage over a 24-hour period. Feature (c) is not used in our detection scheme. We merely include it so that we can compare the characteristics of our dataset with those used by the author in [15]. As we can see form Fig 1(a) the number of followers for legitimate users can be very large but the number of followers for each spammer is typically smaller than that of an average legitimate user. Specifically, the average number of followers for spammers is 4435.7 while that for legitimate users is 7293.1 for our dataset.

From Fig 1(b), we see that the number of "following" for spammers is higher than that for legitimate users. The average number of "following" for spammers is 3535.2 while it is only 1107.7 for legitimate users. Fig 1(c) shows that unlike the plot of reputation (defined in Eqn (1) in Section 3.1) shown in [15], our plots show that the reputation of spammers span a similar range to what is observed for legitimate users and hence reputation metric may not be useful in helping us identify spammers in our dataset. Fig 1(d) shows the average posting percentage over the eight 3-hour interval within a day. The plot clearly shows that normal users tend to tweet during late afternoon while spammers tend to tweet mostly during the early hours.

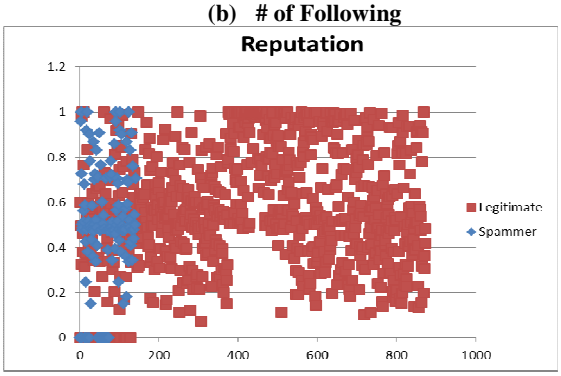
Figs 2(a)-2(d) show the differences between the content-based features of spammers/legitimate users. In Fig 2(a), we see that spammers tend to have an average of 1 link in each of their tweets. As for user mentions shown in Fig 2(b), there are some normal users that carry more user mentions in their tweets. From Fig 2(c), we see that spammers use much more hashtags than normal users. The plot in Fig 2(d) shows that the wordweight for a spammer is usually higher than that of a regular user.



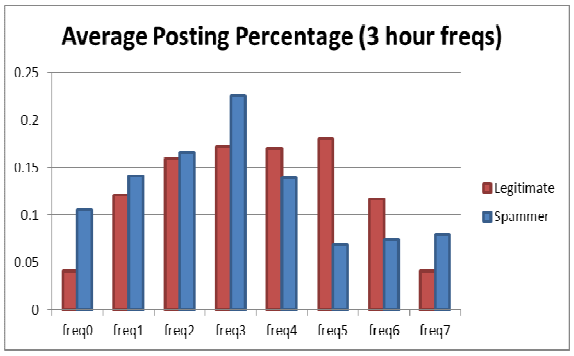
(a) # of Followers



(b) # of Following

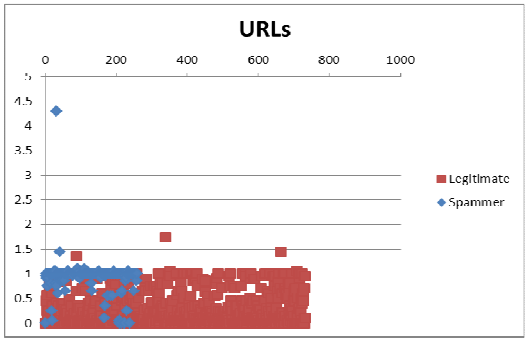


(c) Reputation

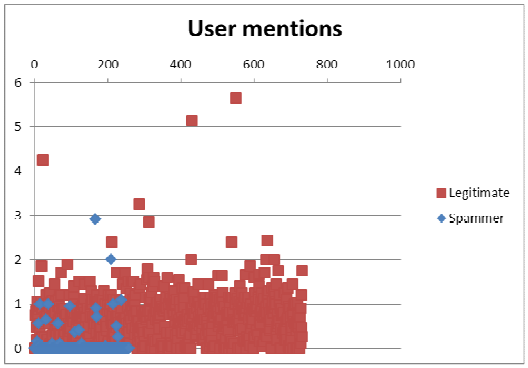


(d) Average Posting Percentage (per 3 hour freqs)

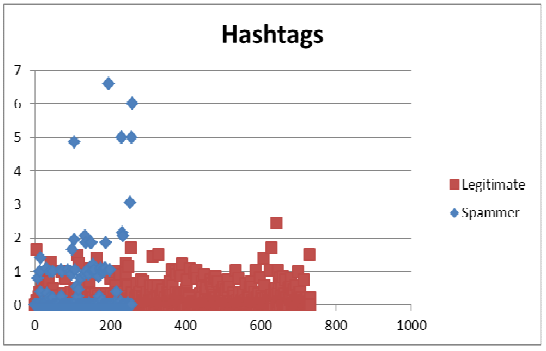
Fig 1: User-Based Features of Spammers/Legitimate Users



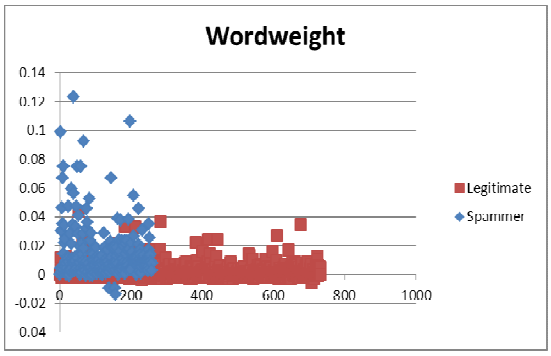
(a) Avg # of URLs



(b) Users/Mentions



(c) # of Hashtags



(d) Wordweight

Fig 2: Characteristics of Content-Based Features of Spammers/Legitimate Users

5. Spam Detection & Evaluations

5.1 Spam Detection

Based on the above identified features, we proceed to use traditional classifiers to help detect spammers. In this work, several classic classification algorithms such as Random Forest, Naïve Bayesian, Support Vector Machines, and K-nearest neighbors are compared. The Random Forest classifier [19] is known to be effective in giving estimates of what variables are important in the classification. This classifier also has methods for balancing error in class population unbalanced data sets.

The naïve Bayesian classifier is based on the well-known Bayes theorem. The big assumption of the naïve Bayesian classifier is that the features are conditionally independent, although research shows that it is surprisingly effective in practice without the unrealistic independence assumption [7]. To classify a data record, the posterior probability is computed for each class [15]:

$$P(Y \mid X) = \frac{P(Y)\prod_{i=1}^{d} P(X_i \mid Y)}{P(X)} \quad (2)$$

Since P(X) is a normalized factor which is equal for all classes, only the numerator needs to be maximized in order to do the classification for the Naïve Bayesian classifier.

The Support Vector Machine method we used is the SMO scheme implemented in the WEKA tool. This SMO scheme, designed by J.C. Platt [16], uses a sequential minimal optimization algorithm to train a support vector classifier using polynomial or RBF kernels. The SMO classifier has been shown to outperform Naives Bayesian classifier in email categorization in [17] when the number of features increases. The K-Nearest Neighbor method implemented in the WEKA tool is the IBK classifier [18].

5.2 Evaluations

We used the standard metrics for measuring the usefulness of our detection scheme that uses our chosen user and

content-based features. The typical confusion matrix for our spam detection system is shown below

		Prediction	
		Spam	Not Spam
True	Spam	a	b
	Not Spam	c	d

where *a* represents the number of spams that were correctly classified, *b* represents the number of spams that were falsely classified as non-spam, *c* represents the number of non-spam messages that were falsely classified as spam, and *d* represents the number of non-spam users that were correctly classified. The following measures are used: precision, recall, and F-measure where the precision is P=a/(a+c), the recall is R=a/(a+b), and the F-measure is defined as F=2PR/(P+R). We have results based on the most recent 20,50 and 100 tweets. Here, we only report the results for the most recent 100 tweets. Our results using the most recent 100 tweets are tabulated in Table 1.

Table 1: Classification Results Using User-Based & Content-Based features (most recent 100 tweets)

Classifier	Precision	Recall	F-measure
RandForest	0.957	0.957	0.957
SMO	0.935	0.931	0.932
NaiveBayes	0.916	0.914	0.915
Ibk(KNN equivalent)	0.928	0.928	0.928

Unlike the results reported in [15], we see that the Random Forest classifier produces the best results, followed by the SMO, Naïve Bayesian and K-NN neighbor classifiers. The good performance of the Random Forest Classifier is not surprising since this classifier can deal with imbalanced data sets (we have data for more regular users than spammers). SMO also has relatively good performance. Naïve Bayesian classifier performs poorer may be because the 100 tweets/user statistics may be noisier than the dataset in [15]. Comparing our results with those reported in [15], we believe that even though we did not use the content similarity feature, our wordweight feature and the percentages of tweet distribution over the 3-hour intervals help our detector to achieve good results.

In Fig 3, we plot the classification results using only user-based features. In Fig 4, we plot the classification results using both user-based and content-based features while in Fig 5, we plot the classification results using all features in Fig 4 except the 8 3-hour interval related tweet distribution features. Comparing Fig 3 with Figs 4 & 5, one can clearly see the benefits of adding the content-based features.

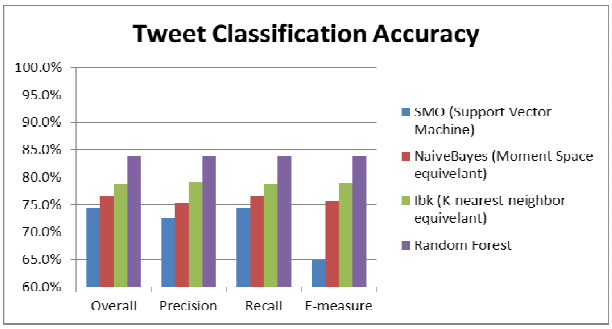


Fig 3: Classification Resutls Using Only User-Based Features with Traditional Classifiers

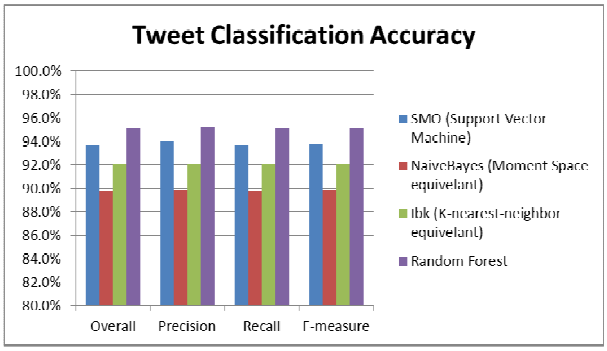


Fig 4: Classification Results Using Both User & Content-Based Features with Traditional Classifiers

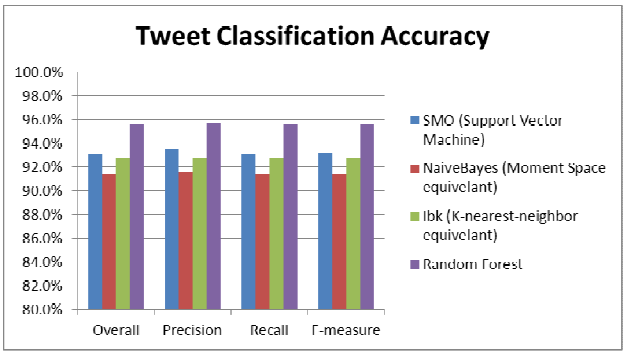


Fig 5: Classification Results Using Both User-Based & Content-Based Features but without the 3-hour interval statistic with Traditional Classifiers

Fig 6 shows the classification results using the features that the researchers describe in [15] with our dataset while Fig 7 shows the classification results using the features that the researchers describe in [14]. Recall that we use wordweight to replace the pairwise content similarity metric. Our results reported in Fig 5 are slightly better than those in Figs 6 & 7. For example, the overall accuracy is only 93.5% with the features suggested in [15], 94.4% with the features suggested in [14] while with our features, we get 95.7% (all with the Random Forest Classifier). Out of the 258 spammers, our detector can correctly classify 240 spammers. Thus, our recall is 93%. Using the features suggested in [14], we can only identify 229 spammers while using the features suggested in [15], we can only identify 230 spammers (89% similar to what they report in their paper).

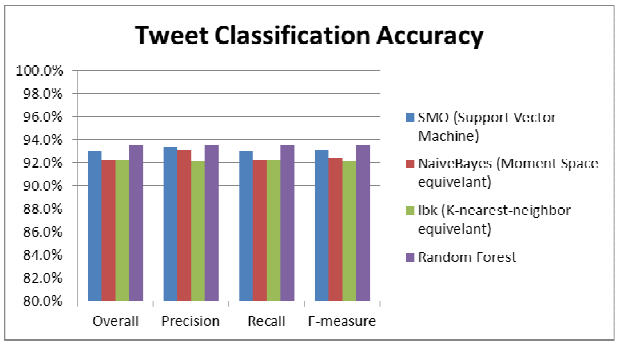


Fig 6: Classification Results using features suggested in [15]

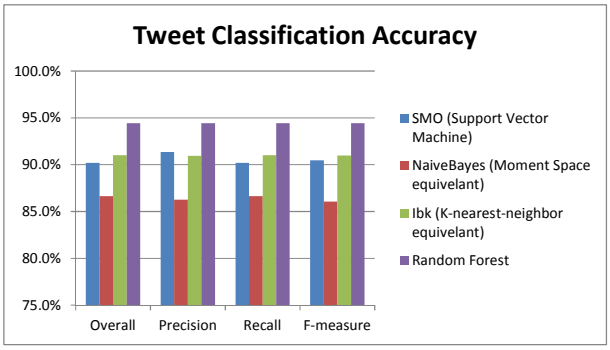


Fig 7: Classification Results using features suggested in [14]

6. Conclusion

In this paper, we have suggested some user-based and content-based features that can be used to distinguish between spammers and legitimate users on Twitter, a popular online social networking site. These suggested features are influenced by Twitter spam policies and our observations of spammers' behaviors. Then, we use these features to help identify spammers. We evaluate the usefulness of these features in spammer detection using traditional classifiers like Random Forest, Naïve Bayesian, Support Vector Machine, K-NN neighbor schemes using the Twitter dataset we have collected. Our results show that the Random Forest classifier gives the best performance. Using this classifier, our suggested features can achieve 95.7% precision and 95.7% F-measure. Based on our dataset, our features provide slightly better classification results when compared to those suggested in [14] or [15]. Our next step is to evaluate our detection scheme using larger Twitter dataset as well as possibly wall-post datasets from other online networking sites like Facebook. We also hope to include the content similarity metric in our near future work.

7. REFERENCES

[1] M. Mowbray, "The Twittering Machine", Proceedings of the 6th International Conference on Web Information and Technologies, April 2010.

[2] Analytics, P., "Twitter study- August 2009", http://www.peranalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009.pdf

[3] CNET (2009). 4 chan may be behind attack on twitter. http://news.cnet.com/8301-13515_3-10279618-26.html.

[4] How to; 5 Top methods & applications to reduce Twitter Spam http://blog.thoughtpick.com/2009/07/how-to-5-top-methods-applications-to-reduce-twitter-spam.html

[5] Twitter (2009a), "Restoring accidentally suspended accounts.",http://status.twitter.com/post/136164828/restoring-accidentally-suspended-accounts.

[6] Twitter (2009b). The twitter rules. http://status.twitter.com/post/136164828/restoring-accidentally-suspended-accounts.

[7] I. Rish. "An empirical study of the naïve bayes classifier". Proeedings of IJCAI workshop on Empirical Methods in Artificial Intelligence, 2005.

[8] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3 (Mar. 2003), 1289-1305.

[9] Compete site comparison http://siteanalytics.compete.com/facebook.com+myspace.com+twitter.com/

[10] Sophos facebook id probe, http://www.sophos.com/pressoffice/news/articles/2007/08/facebook.html, 2008.

[11] L. Bilge et al, "All your contacts are belong to us: automated identifty theft attacks on social networks", Proceedings of ACM World Wide Web Conference, 2009.

[12] T.N. Jagatic et al, "Social Phishing", Communications of ACM, Vol 50(10):94-100, 2007.

[13] S. Yardi et al, "Detecting Spam in a Twitter Network", First Monday, Vol 15(1), 2010.

[14] G. Stringhini, C. Kruegel, G. Vigna, "Detecting Spammers on Social Networks", Proceedings of ACM ACSAS'10, Dec, 2010.

[15] A. H. Wang, "Don't Follow me: Twitter Spam Detection", Proceedings of 5th International Conference on Security and Cryptography, July, 2010.

[16] J. Platt, "Sequential Minimal Optimization: A fast algorithm for training support vector machines", Advanced in Kernel Methods – Support Vector learning, B. Schoelkopf et al, eds, MIT Press.

[17] H. Berger, M. Kohle, D. Merkl, "On the impact of document representation on classifier performance in email categorization", Proceedings of the 4th International Conference on Information Systems Technology and IST Applications, May, 2005.

[18] D. Aha, D. Kibler, "Instance-based Learning Algorithms", Machine Learning, Vol 6, pp 37-66.

[19] L. Breiman, "Random Forests", Machine Learning, Vol 45, Issue 1, Oct, 2001.