Detecting Spammers on Twitter

Fabrício Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgílio Almeida
Computer Science Department, Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
{fabricio, magno, tiagorm, virgilio}@dcc.ufmg.br

ABSTRACT

With millions of users tweeting around the world, real time search systems and different types of mining tools are emerging to allow people tracking the repercussion of events and news on Twitter. However, although appealing as mechanisms to ease the spread of news and allow users to discuss events and post their status, these services open opportunities for new forms of spam. Trending topics, the most talked about items on Twitter at a given point in time, have been seen as an opportunity to generate traffic and revenue. Spammers post tweets containing typical words of a trending topic and URLs, usually obfuscated by URL shorteners, that lead users to completely unrelated websites. This kind of spam can contribute to de-value real time search services unless mechanisms to fight and stop spammers can be found.

In this paper we consider the problem of detecting spammers on Twitter. We first collected a large dataset of Twitter that includes more than 54 million users, 1.9 billion links, and almost 1.8 billion tweets. Using tweets related to three famous trending topics from 2009, we construct a large labeled collection of users, manually classified into spammers and non-spammers. We then identify a number of characteristics related to tweet content and user social behavior, which could potentially be used to detect spammers. We used these characteristics as attributes of machine learning process for classifying users as either spammers or non-spammers. Our strategy succeeds at detecting much of the spammers while only a small percentage of non-spammers are misclassified. Approximately 70% of spammers and 96% of non-spammers were correctly classified. Our results also highlight the most important attributes for spam detection on Twitter.

Keywords: spam, twitter, real time search, spammer, microblogging, online social networks, machine learning.

1. INTRODUCTION

Twitter has recently emerged as a popular social system where users share and discuss about everything, including news, jokes, their take about events, and even their mood. With a simple interface where only 140 character messages can be posted, Twitter is increasingly becoming a system for obtaining real time information. When a user posts a tweet, it is immediately delivered to her followers, allowing them to spread the received information even more. In addition to be received by followers, tweets can also be retrieved through search systems and other tools. With the emergence of real time search systems and meme-tracking services, the repercussion of all kinds of events and news are beginning to be registered with practically no delay between the creation and availability for retrieval of content. As example, Google, Bing, Twitter and other meme-tracking services are mining real time tweets to find out what is happening in the world with minimum delay [4].

However, although appealing as mechanisms to ease the spread of news and allow users to discuss events and post their status, these services also open opportunities for new forms of spam. For instance, trending topics, the most talked about items on Twitter at a given point in time, have been seen as an opportunity to generate traffic and revenue. When noteworthy events occur, thousands of users tweet about it and make them quickly become trending topics. These topics become the target of spammers that post tweets containing typical words of the trending topic, but URLs that lead users to completely unrelated websites. Since tweets are usually posted containing shortened URLs, it is difficult for users to identify the URL content without loading the webpage. This kind of spam can contribute to reduce the value of real time search services unless mechanisms to fight and stop spammers can be found.

Tweet spammers are driven by several goals, such as to spread advertise to generate sales, disseminate pornography, viruses, phishing, or simple just to compromise system reputation. They not only pollute real time search, but they can also interfere on statistics presented by tweet mining tools and consume extra resources from users and systems. All in all, spam wastes human attention, maybe the most valuable resource in the information age.

Given that spammers are increasingly arriving on Twitter, the success of real time search services and mining tools relies at the ability to distinguish valuable tweets from the spam storm. In this paper, we firstly address the issue of detecting spammers on Twitter. To do it, we propose a 4-step approach. First, we crawled a near-complete dataset from Twitter, containing more than 54 million users, 1.9 billion links, and almost 1.8 billion tweets. Second, we created a labeled collection with users "manually" classified as spammers and non-spammers. Third, we conducted a study about the characteristics of tweet content and user behavior on Twitter aiming at understanding their relative discriminative power to distinguish spammers and non-spammers. Lastly, we investigate the feasibility of applying a super-

CEAS 2010 - Seventh annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference July 13-14, 2010, Redmond, Washington, US

vised machine learning method to identify spammers. We found that our approach is able to correctly identify the majority of the spammers (70%), misclassifying only 3.6% of non-spammers. We also investigate different tradeoffs for our classification approach namely, the attribute importance and the use of different attribute sets. Our results show that even using different subsets of attributes, our classification approach is able to detect spammers with high accuracy. We also investigate the detection of spam instead of spammers. Although results for this approach showed to be competitive, the spam classification is more susceptible to spammers that adapt their strategies since it is restricted to a small and simple set of attributes related to characteristics of tweets.

The rest of the paper is organized as follows. The next section presents a background on Twitter and provides the definition of spam used along this work. Section 3 describes our crawling strategy and the labeled collection built from the crawled dataset. Section 4 investigates a set of user attributes and their ability to distinguish spammers and non-spammers. Section 5 describes and evaluates our strategies to detect spammers and Section 6 surveys related work. Finally, Section 7 offers conclusions and directions for future work.

2. BACKGROUND AND DEFINITIONS

Twitter is an information sharing system, where users follow other users in order to receive information along the social links. Such information consists of short text messages called tweets. Relationship links are directional, meaning that each user has followers and followees, instead of unidirectional friendship links. Tweets can be repeated throughout the network, a process called re-tweeting. A retweeted message usually starts with "RT @username", where the @ sign represents a reference to the one who originally posted the messages. Twitter users usually use hashtags (#) to identify certain topics. Hashtags are similarly to a tag that is assigned to a tweet in its own body text.

The most popular hashtags or key words that appear on tweets become trending topics. Most of the trending topics reflect shocking and breaking news or events that appear in the mass media. Among the most popular events in 2009 that also became popular trending topics are Michael Jackson's death, Iran election, and the emergence of the British singer, Susan Boyle, on the TV show *Britain's Got Talent* [2].

However, the most popular hashtag recorded in 2009 is not related to news or events that appeared in the traditional mass media. The hashtag #musicmonday is widely used by users to weekly announce tips about music, songs, or concerts. Several users post what kind of song they are listening to every Monday and add that hashtag so that others can search. Such hashtags are conventions created by users that become largely adopted. As example, the first tweet in our dataset with this hashtag says:

What are you listening to? Tag it, #musicmonday "Come Together"- The Beatles.

Figure 1 shows part of the results of a search on Twitter for the hashtag #musicmonday. The figure shows three tweets that appear as result and contains most of the elements we discussed here. We can note on the figure a list

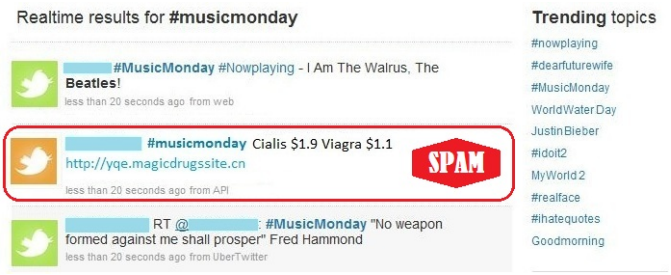


Figure 1: Ilustrative example of a search on Twitter for the hashtag #musicmonday

of trending topics, hashtags, retweets, and anonymized user names. The second tweet is an example of a tweet spam, since it contains a hashtag completely unrelated to the URL the tweet points to. In this paper, we consider as spammers on Twitter those users who post at least *one* tweet containing a URL considered unrelated to the tweet body text. Examples of tweet spam are: (i) a URL to a website containing advertisements completely unrelated to a hashtag on the tweet, and (ii) retweets in which legitimate links are changed to illegitimate ones, but are obfuscated by URL shorteners.

Although there are other forms of opportunistic actions in Twitter, not all of them can be considered as spam. As example, there are opportunistic users that follow a large number of people in an attempt to be followed back and then disseminate their messages. Here we do not consider content received through the social links as spam since users are free to follow the users they want.

3. DATASET AND LABELED COLLECTION

In order to evaluate our approach to detect spammers on Twitter, we need a labeled collection of users, pre-classified into spammers and non-spammers. To the best of our knowledge, no such collection is publicly available. We then had to build one. Next, we describe the strategy used to collect Twitter in Section 3.1. We then discuss the process used to select and manually classify a subset of spammers and non-spammers in Section 3.2.

3.1 Crawling twitter

In analyzing the characteristics of users in Twitter, ideally we would like to have at our disposal data for each existing Twitter user, including their social connections, and all the tweets they ever posted. So, to that end, we asked Twitter to allow us to collect such data and they white-listed 58 servers located at the Max Planck Institute for Software Systems (MPI-SWS), located in Germany[1]. Twitter assigns each user a numeric ID which uniquely identifies the user's profile. We launched our crawler in August 2009 to collect all user IDs ranging from 0 to 80 million. Since no single user in the collected data had a link to a user whose ID is greater than 80 million, our crawler has inspected all users with an account on Twitter. In total, we found **54,981,152** used accounts that were connected to each other by **1,963,263,821** social links. We also collected all tweets ever posted by the collected users, which consists of a total of **1,755,925,520**

[1]Part of this work was done when the first author was visiting the MPI-SWS

tweets. Out of all users, nearly 8% of the accounts were set private, so that only their friends could view their tweets. We ignore these users in our analysis. The link information is based on the final snapshot of the network topology at the time of crawling and we do not know when the links were formed. We plan to make this data available to the wider community. For a detailed description of this dataset we refer the user to our project homepage [3].

3.2 Building a labeled collection

Next, we describe the steps taken to build our labeled collection. There are three desired properties that need to be considered to create such collection of users labeled as spammers and non-spammers. First, the collection needs to have a significant number spammers and non-spammers. Second, the labeled collection needs to include, but not restricting to, spammers who are aggressive in their strategies and mostly affect the system. Third, it is desirable that users are chosen randomly and not based on their characteristics.

In order to meet these three desired properties, we focus on users that post tweets about three trending topics largely discussed in 2009. (1) the Michael Jackson's death, (2) Susan Boyle's emergence, and (3) the hashtag "#musicmonday". Table 1 summarizes statistics about the number of tweets we have in our dataset as well as the number of unique users that spread these tweets. We obtained a key date for the event related to Susan Boyle and Michael Jackson; this either corresponds to the date when the event occurred was widely reported in the traditional mass media (TV and news papers) until the last day they appear in our data. For the #musicmonday we used all tweets with the hashtag. Figure 2(a) shows an example of the daily frequency of tweets about #musicmonday accross a two month period. We can note a clearly week pattern with strong peaks on Mondays. The weekly popularity of this hashtag made it become a popular topic accross most of 2009 and the most popular in terms of number of tweets. On the other hand, news and events have a different pattern with most of the popularity concentrated around the days of the event. Figure 2(b) shows peaks on events related to Michael Jackson's death and Figure 2(c) shows peaks around Susan Boyle's performance on the TV show. Table 1 summarizes statistics about the amount of data used for each event.

By choosing these events, we include spammers that are aggressive in their strategies and target trending topics. Aiming at capturing the other two desired properties, we randomly selected users among the ones that posted at least one tweet containing a URL with at least one key work described in Table 1. Then, we developed a website to help volunteers to manually label users as spammers or non-spammers based on their tweets containing #keywords related to the trending topics. In order to minimize the impact of human error, two volunteers analyzed each user in order to independently label her or him as spammer or non-spammer. In case of tie (i.e., each volunteer chooses a class), a third independent volunteer was heard. Each user was classified based on majority voting. Volunteers were instructed to favor non-spammers in case of doubt. For instance, if one was not confident that a tweet was unrelated to music, she should consider it to be non-spammer. The volunteers agreed in almost 100% of the analyzed tweets, which reflects a high level of confidence to this human classification process.

In total, 8,207 users were labeled, including 355 spammers and 7,852 non-spammers. Since the number of non-spammers is much higher than the number of spammers, we randomly select only 710 of the legitimate users to include in our collection, which corresponds to twice the number of spammers. Thus, the total size of our labeled collection is 1,065 users. Since the user classification labeling process relies on human judgment, which implies in reading a significantly high amount of tweets, we had to set a limit on the number of users in our labeled collection. Among the forms of spam found, our volunteers reported a number of websites containing pornography, advertisements, phishing, and even executable files. We plan to make our labeled collection available to the research community in due time.

4. IDENTIFYING USER ATTRIBUTES

Unlike common Twitter users, people who spam usually aim at commercial intent (e.g., advertising) and belittlement of ideas and system reputation [17]. Since non-spammers and spammers have different goals in the system, we expect they also differ on how they behave (e.g., who they interact with, which frequency they interact, etc.) to achieve their purposes. Intuitively, we expect that non-spammers spend more time interacting with other users, doing actions like replying, retweeting, posting status without URL, etc. In order to verify this intuition, we looked at the characteristics of the users of the labeled collection. We analyze a large set of attributes that reflect user behavior in the system as well as characteristics of the content posted by users, aiming at investigating their relative discriminatory power to distinguish one user class from the other. We considered two attribute sets, namely, content attributes and user behavior attributes, discussed next.

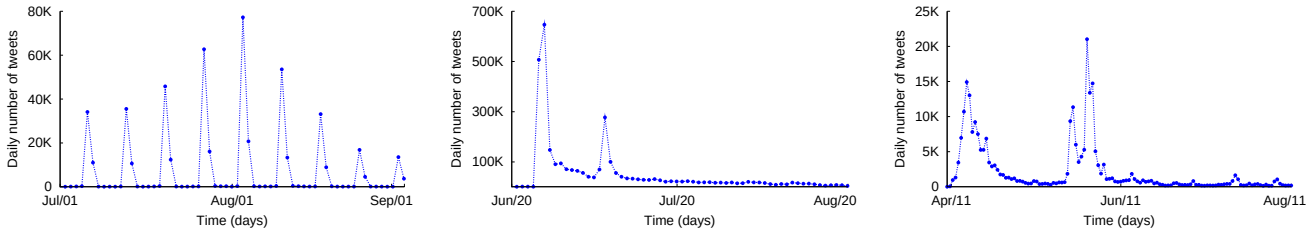
4.1 Content attributes

Content attributes are properties of the text of tweets posted by users, which capture specific properties related to the way users write tweets. Given that users usually post several tweets, we analyze tweet content characteristics based on the maximum, minimum, average, and median of the following metrics: number of hashtags per number of words on each tweet, number of URLs per words, number of words of each tweet, number of characters of each tweet, number of URLs on each tweet, number of hashtags on each tweet, number of numeric characters (i.e. 1,2,3) that appear on the text, number of users mentioned on each tweet, number of times the tweet has been retweeted (counted by the presence of "RT @username" on the text). We also considered the fraction of tweets with at least one word from a popular list of spam words [1], the fraction of tweets that are reply messages, and the fraction of tweets of the user containing URLs. In total, we have 39 attributes related to content of the tweets.

Next, we look into three characteristics of the tweet content that can differ spammers from non-spammers. Figure 3 shows the cumulative distribution function (CDF) for three content attributes: fraction of tweets containing URLs, fraction of tweets that contains spam words, and average number of words that are hashtags on the tweet. We notice from Figure 3 (a) that spammers do post a much higher fraction of tweets with URLs, compared to non-spammers. Naturally, spammers also post a much larger portion of their tweets containing spam words than non-spammers, as we can see on Figure 3 (b). For example, 39% of the spammers posted all

Topic	Period	Keywords	Tweets	Users
#musicmonday	Dec 8,2008—Sep 24,2010	#musicmonday	745,972	183,659
Boyle	April 10—Sep 24,2010	"Susan Boyle", #susanboyle	264,520	146,172
Jackson	Jun 25—Sep 24,2010	"Michael Jackson", #michaeljackson, #mj	3,184,488	1,232,865

Table 1: Summary information of three events considered to construct the labeled collection



(a) #musicmonday (b) Michael Jackson (c) Susan Boyle

Figure 2: Daily number of tweets for the three events analyzed

their tweets containing spam words, whereas non-spammers typically do not post more than 4% of their tweets containing spam word. This huge discrepancy also reflects the early stage of the spamming process on Twitter. Although a single mechanisms like a spam word mechanism could filter most of the spam tweets posted today, such metric can be easily manipulated by spammers. The last attribute we analyze is the average fraction of hashtags per tweet posted per user. Figure 3 (c) shows the CDF for this metric. As expected, spammers post a higher fraction of hashtags per tweet. We noted that in our labeled collection some spammers post a large number of popular hashtags, spanning a large number of different trending topics within a single tweet. In general, the analysis of these attributes show that characteristics of the tweet content have potential to differentiate spammers from non-spammers.

4.2 User behavior attributes

User attributes capture specific properties of the user behavior in terms of the posting frequence, social interactions, and influence on the Twitter network. We considered the following metrics as user attributes: number of followers, number of followees, fraction of followers per followees, number of tweets, age of the user account, number of times the user was mentioned, number of times the user was replied to, number of times the user replied someone, number of followees of the user's followers, number tweets receveid from followees, existence of spam words on the user's screename, and the minimum, maximum, average, and median of the time between tweets, number of tweets posted per day and per week. In total, we have 23 attributes about the user behavior.

Next, we show in detail three characteristics of user behavior: the number of followers per number of followees, the age of the user account, and the number of tweets received. Figure 4 shows the CDF for these attributes. We can clearly note by Figure 4 (a) that spammers have a high ratio of followers per followees in comparison with non-spammers. Spammers try to follow a large number of users as attempt to be followed back, which does not happen for most of the cases. This behavior makes the fraction of followers per followees very small for spammers. Figure 4 (b) shows the age of the user account. Spammers usually have new accounts probably because they are constantly being blocked by other users and reported to Twitter. Lastly, we look at the number of tweets posted by the followees of the spammers. Figure 4 (c) shows that non-spammers receive a much large amount of tweets from their followees in comparison with spammers. Some spammers do not even follow other users and just focus on quickly post spamming after the account is created.

Other metrics such as the number of times the user was mentioned by other users and number of times the user was replied can be useful to differentiate spammers and promoters, since they capture the notion of influence of the users in the Twitter network [11].

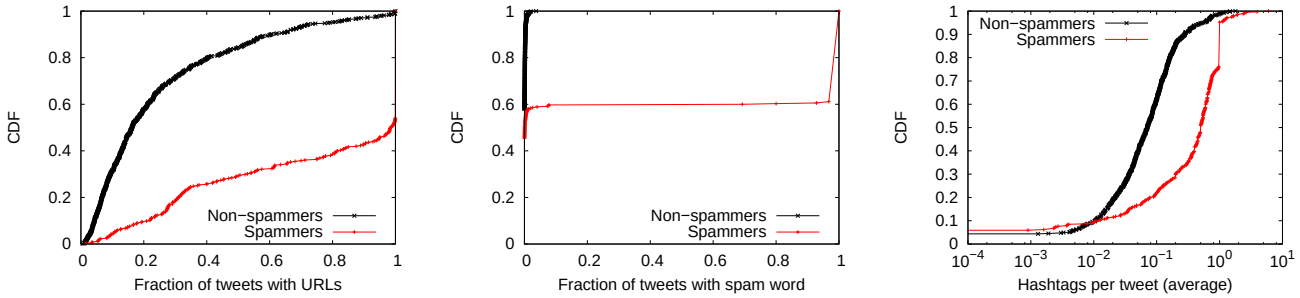
5. DETECTING SPAMMERS

In this section, we investigate the feasibility of applying a supervised learning algorithm along with the attributes discussed in the previous section for the task of detecting spammers on Twitter. In this approach, each user is represented by a vector of values, one for each attribute. The algorithm learns a classification model from a set of previously labeled (i.e., pre-classified) data, and then applies the acquired knowledge to classify new (unseen) users into two classes: spammers and non-spammers. Note that, in this paper, we created a labeled collection. In a practical scenario, labeled data may be obtained through various initiatives (e.g., volunteers who help marking spam, professionals hired to periodically manually classify a sample of users, etc). Our goal here is to assess the *potential effectiveness* of the proposed approach as a first effort towards detecting spammers.

We continue by presenting, in Section 5.1, the metrics used to evaluate our experimental results. Section 5.2 describes the classification algorithm, i.e., the classifier, and the experimental setup used.

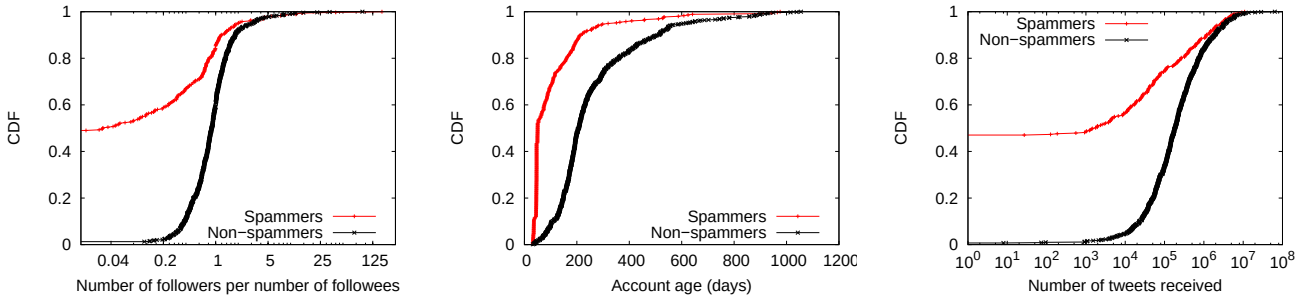
5.1 Evaluation metrics

To assess the effectiveness of our classification strategies we use the standard information retrieval metrics of recall, precision, Micro-F1, and Macro-F1 [30]. The recall (r) of a class X is the ratio of the number of users correctly classified to the number of users in class X. Precision (p) of a class X



(a) Fraction of tweets containing URLs (b) Fraction of tweets with spam words (c) Average number of hashtags per tweet

Figure 3: Cumulative distribution functions of three content attributes



(a) Fraction of followers per followees (b) Age of the user account (c) Number of tweets received

Figure 4: Cumulative distribution functions of three user behavior attributes

is the ratio of the number of users classified correctly to the total predicted as users of class X. In order to explain these metrics, we will make use of a confusion matrix [20], illustrated in Table 2. Each position in this matrix represents the number of elements in each original class, and how they were predicted by the classification. In Table 2, the precision (p_{spam}) and the recall (r_{spam}) indices of class spammer are computed as $p_{spam} = a/(a+c)$ and $r_{spam} = a/(a+b)$.

		Predicted	
		Spammer	**Non-spammer**
True	**Spammer**	a	b
	Non-spammer	c	d

Table 2: Example of confusion matrix

The F1 metric is the harmonic mean between both precision and recall, and is defined as $F1 = 2pr/(p+r)$. Two variations of F1, namely, micro and macro, are usually reported to evaluate classification effectiveness. Micro-F1 is calculated by first computing global precision and recall values for all classes, and then calculating F1. Micro-F1 considers equally important the classification of *each user*, independently of its class, and basically measures the capability of the classifier to predict the correct class on a per-user basis. In contrast, Macro-F1 values are computed by first calculating F1 values for each class in isolation, as exemplified above for spammers, and then averaging over all classes. Macro-F1 considers equally important the effectiveness in *each class*, independently of the relative size of the class. Thus, the two metrics provide complementary assessments of the classification effectiveness. Macro-F1 is especially important when the class distribution is very skewed, as in our case, to verify the capability of the classifier to perform well in the smaller classes.

5.2 The classifier and the experimental setup

We use a Support Vector Machine (SVM) classifier [19], which is a state-of-the-art method in classification and obtainned the best results among a set of classifiers tested. The goal of a SVM is to find the hyperplane that optimally separates with a maximum margin the training data into two portions of an N-dimensional space. A SVM performs classification by mapping input vectors into an N-dimensional space, and checking in which side of the defined hyperplane the point lies. We use a non-linear SVM with the Radial Basis Function (RBF) kernel to allow SVM models to perform separations with very complex boundaries. The implementation of SVM used in our experiments is provided with libSVM [13], an open source SVM package that allows searching for the best classifier parameters using the *training* data, a mandatory step in the classifier setup. In particular, we use the *easy* tool from libSVM, which provides a series of optimizations, including normalization of all numerical attributes. For experiments involving the SVM J parameter (discussed in Section 5.3), we used a different implementation, called SVM light, since libSVM does not provide this parameter. Classification results are equal for both implementations when we use the same classifier parameters.

The classification experiments are performed using a 5-fold cross-validation. In each test, the original sample is partitioned into 5 sub-samples, out of which four are used as training data, and the remaining one is used for testing

the classifier. The process is then repeated 5 times, with each of the 5 sub-samples used exactly once as the test data, thus producing 5 results. The entire 5-fold cross validation was repeated 5 times with different seeds used to shuffle the original data set, thus producing 25 different results for each test. The results reported are averages of the 25 runs. With 95% of confidence, results do not differ from the average in more than 5%.

5.3 Basic classification results

Table 3 shows the confusion matrix obtained as the result of our experiments with SVM. The numbers presented are percentages relative to the total number of users in each class. The diagonal in boldface indicates the recall in each class. Approximately, 70% of spammers and 96% of non-spammers were correctly classified. Thus, only a small fraction of non-spammers were erroneously classified as spammers.

A significant fraction (almost 30%) of spammers was misclassified as non-spammers. We noted that, in general, these spammers exhibit a dual behavior, sharing a reasonable number of non-spam tweets, thus presenting themselves as non-spammers most of the time, but occasionally some tweet that was considered as spam. This dual behavior masks some important aspects used by the classifier to differentiate spammers from non-spammers. This is further aggravated by the fact that a significant number of non-spammers post their tweets to trending topics, a typical behavior of spammers. Although the spammers our approach was not able to detect are occasional spammers, an approach that allow one to choose to detect even occasional spammers could be of interest. In Section 5.4, we discuss an approach that allows one to trade a higher recall of spammers at a cost of misclassifying a larger number of non-spammers.

		Predicted	
		Spammer	**Non-spammers**
True	**Spammer**	**70.1%**	29.9%
	Non-spammer	3.6%	**96.4%**

Table 3: Basic classification results

As a summary of the classification results, Micro-F1 value is 87.6, whereas per-class F1 values are 79.0 and 91.2, for spammers and non-spammers, respectively, resulting in an average Macro-F1 equal to 85.1. The Micro-F1 result indicates that we are predicting the correct class in 87.6% of the cases. Complementarily, the Macro-F1 result shows that there is a certain degree of imbalance for F1 across classes, with more difficulty for classifying spammers. Comparing with a trivial baseline classifier that chooses to classify every single user as non-spammer, we obtain gains of about 31.4% in terms of Micro-F1, and of 112.8% in terms of Macro-F1.

5.4 Spammer detection tradeoff

Our basic classification results show we can effectively identify spammers, misclassifying only a small fraction of non-spammers. However, even the small fraction of misclassified non-spammers could not be suitable for a detection mechanism that apply some sort of automatic punishment to users. Additionally, one could prefer identifying more spammers at the cost of misclassifying more non-spammers.

This tradeoff can be explored using a cost mechanism, available in the SVM classifier. In this mechanism, one can give priority to one class (e.g., spammers) over the other (e.g., non-spammers) by varying its J parameter[2] [24].

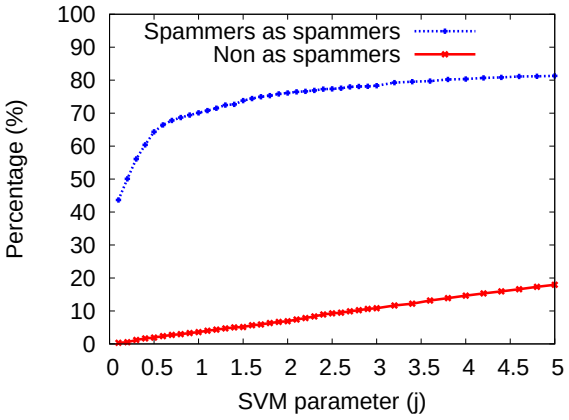


Figure 5: Impact of varying the J parameter

Figure 5 shows classification results when we vary the parameter J. We can note that increasing J leads to a higher percentage of correctly classified spammers (with diminishing returns for $J > 0.3$), but at the cost of a larger fraction of misclassified legitimate users. For instance, one can choose to correctly classify around 43.7% of spammers, misclassifying only 0.3% non-spammers ($J = 0.1$). On the other hand, one can correctly classify as much as 81.3% of spammers ($J = 5$), paying the cost of misclassifying 17.9% of legitimate users. The best solution to this tradeoff depends on the system's objectives. For example, a system might be interested in sending an automatic warning message to all users classified as spammers, in which case they might prefer to act conservatively, avoiding sending the message to legitimate users, at the cost of reducing the number of correctly predicted spammers. In another situation, a system may prefer to filter any spam content and then detect a higher fraction of spammers, misclassifying a few more legitimate users. It should be stressed that we are evaluating the potential benefits of varying J. In a practical situation, the optimal value should be discovered in the training data with cross-validation, and selected according to the system's goals.

5.5 Importance of the attributes

In order to verify the ranking of importance of these attributes we use two feature selection methods available on Weka [27]. We assessed the relative power of the 60 selected attributes in discriminating one user class from the others by independently applying two well known feature selection methods, namely, information gain and χ^2 (Chi Squared) [31]. Since results for information gain and χ^2 are very similar and both methods ranked 10 attributes in common among the top 10, we omitted results for information gain. Table 4 presents the 10 most important attributes for the χ^2 method.

We can note that two of the most important attributes are the fraction of tweets with URLs and the average number

[2]The J parameter is the cost factor by which training errors in one class outweigh errors in the other. It is useful, when there is a large imbalance between the two classes, to counterbalance the bias towards the larger one.

Position	χ^2 ranking
1	fraction of tweets with URLs
2	age of the user account
3	average number of URLs per tweet
4	fraction of followers per followees
5	fraction of tweets the user had replied
6	number of tweets the user replied
7	number of tweets the user receive a reply
8	number of followees
9	number of followers
10	average number of hashtags per tweet

Table 4: Ranking of the top 10 attributes

of URLs per tweet. Although these attributes are redundant, the importance of them highlight an interesting aspect of spammers. Spammers are most interested in spreading advertisements that usually points to an website istead of spreading rumors or an specific piece of message. Thus, spammers usually post URLs whereas non-spammers post a number of status updates without URLs. We can also note that spammers are usually associated with new accounts. Thus, ignore tweets from very new accounts on results of search or mining tools can be a nice strategy to avoid spam.

	Tweet content	User behavior
Top 10	4	6
Top 20	10	10
Top 30	17	13
Top 40	23	17
Top 50	31	19
Top 62	39	23

Table 5: Number of attributes at top positions in the χ^2 ranking

Table 5 summarizes the results, showing the number of attributes from each set (tweet content and user behavior) in the top 10, 20, 30, 40, 50 and 62 most discriminative attributes according to the ranking produced by χ^2. Note that, both content attributes and user behavior attributes appear balanced along the entire rank. Given that content attributes are easy to be changed by spammers, such homogeneity means that attributes that are not so easy to be manipulated by spammers could be used instead.

Once we have understood the importance of the attributes used, we now turn to investigate whether competitive effectiveness can be reached with fewer attributes or different sets of attributes.

5.6 Reducing the attribute set

The detection of spammers on Twitter is a form of adversarial fight between spammers and anti-spammers mechanisms. In the long term, it is expect that spammers will evolve and adapt to anti-spammers strategies (i.e. using fake accounts to forge some attributes) [12]. Consequently, some attributes may become less important whereas others may acquire importance with time. Thus, it is important to understand if different sets of attributes could lead our approach to accurate classification results.

Next, we compute the classification results considering different subsets of 10 attributes that occupy contiguous positions in the ranking (i.e., the first top 10 attributes, the next 10 attributes, etc) are used. Figure 6 shows Micro-F1 and Macro-F1 values for the basic classification for the χ^2.

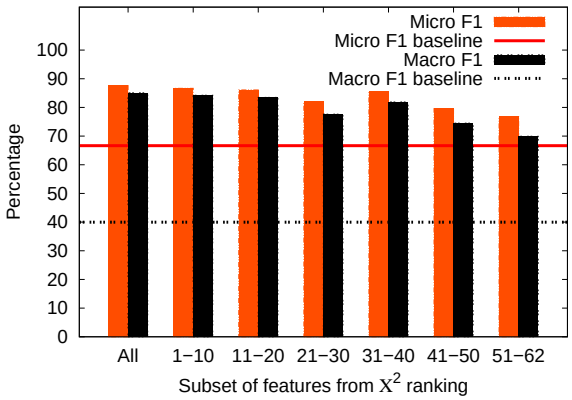


Figure 6: Classification results with groups of ranked attributes according to the χ^2 feature selection algorithm

We compare results with a baseline classifier that considers all users as non-spammers, for each such range. In terms of Micro-F1, our classification provides gains over the baseline for the first two subsets of attributes, whereas significant gains in Macro-F1 are obtained for all attribute ranges, but the last one (the 10 worst attributes). This confirms the results of our attribute analysis that shows that even low-ranked attributes have some discriminatory power. In practical terms, significant improvements over the baseline are possible even if not all attributes considered in our experiments can be obtained.

5.7 Detecting tweets instead of users

Our approach for the spam problem on Twitter focuses on the detection of spammers instead of tweets containing spam. The detection of the spam itself can be useful for filtering spam on real time search whereas the detection of spammers is more associated with the detection of existent spam accounts. Once a spammer is detected, it is natural to suspend her account or even block IP addresses temporarily to prevent spammers from continuing posting spam with new accounts.

Here, we briefly investigate an approach to detect spam instead of the spammers. We consider the following attributes for each tweet: number of words from a list of spam words, number of hashtags per words, number of URLs per words, number of words, number of numeric characters on the text, number of characters that are numbers, number of URLs, number of hashtags, number of mentions, number of times the tweet has been replied (counted by the presence of "RT @username" on the text), and lastly we verified if the tweet was posted as a reply.

Table 6 shows the resulting confusion matrix obtained from the SVM classifier when we use as labeled collection, the tweets classified as spam and non-spam. We can note that approximately 78.5% of spam and 92.5% of the non-spam tweets were correctly classified. Although we are able to misclassify less spam in comparison to our basic classification of spammers, about 7.5% of the non-spam tweets were classified as spam. This happens because for the spammer detection problem, some user present a dual behavior, a problem that we do not have with the classification of tweets.

However, when users post non-spam tweets containing suspect content, i.e. spam words, more than two hashtags, etc., the classifier can make mistakes.

In terms of accuracy (Micro F1), results for both classification strategies are very similar: 87.2% for spam detection and 87.6% for spammer detection. Given that the metrics used for the classification of spam are based only on the tweet content, they could be more easily manipulated by spammers. Although it is useful to have simple forms of spam detection in real time search systems, other techniques are equally important. In a scenario where spammers evolve their detection strategies and manipulate tweet content to make it look like a common tweet, simple detection schemes would fail.

		Predicted	
		Spam	**Non-spam**
True	**Spam**	**78.5%**	21.5%
	Non-spam	92.5%	**7.5%**

Table 6: Detection of spam instead of spammers

In Table 7 we show the results for detection of spammers without considering any metric related to the tweet content. We can note that even removing all attributes related to the content of tweets, we are still able to find spammer accounts with reasonable accuracy (84.5%), using only the attributes related to user behavior.

		Predicted	
		Spammer	**Non-spammers**
True	**Spammer**	**69.7%**	30.3%
	Non-spammer	4.3%	**95.7%**

Table 7: Impact on spammer detection results when removing attributes related to tweets

6. RELATED WORK

Spam has been observed in various applications, including e-mail [9], Web search engines [14], blogs [25], videos [7, 8], and opinions [18]. Consequently, a number of detection and combating strategies have been proposed [16, 22, 29]. Particularly, there has been a considerable number of efforts that rely on machine learning to detect spam. Castillo *et al.* [10] proposed a framework to detect Web spamming which uses a classification approach and explore social network metrics extracted from the Web graph. Similarly, Benevenuto *et al.* [6] approached the problem of detecting spammers on video sharing systems. By using a labeled collection of users manually classified, they applied a hierarchical machine learning approach to differentiate opportunistic users from the non-opportunistic ones in video sharing systems. Classification has also showed to be efficient to detect image-based email that contains spam [5, 28].

Another interesting approach to prevent spam consists of white-listing users so that each user specifies a list of users who they are willing to receive content from. "RE" [15] is a white-listing system for email based on social links that allows emails between friends and friends-of-friends to bypass standard spam filters. Socially-connected users provide secure attestations for each others' email messages while keeping users' contacts private. More recently, Mislove *et al.* [23] propose Ostra, a mechanism that imposes an upfront cost to senders for each communication. Our approach is complementary to Ostra, since we focused on dynamically detecting the originators of spam messages on real time search and Ostra is focused on making the life of originators of messages harder as a form to prevent the problem.

There has been a few concurrent work that reported the existence of spam on Twitter. Kuak *et al.* [21] has reported spam on the twitter data they collected. In order to filter spam and proceed with their analysis, they filter tweets from users who have been on Twitter for less than a day as well as tweets that contain three or more trending topics. Indeed, in our work we have observed that these two characteristics represent important attributes to different spammers from non-spammers. However, our strategy uses a larger set of other attributes and a machine learning technique instead of fixed thresholds. Yard *et al.* [32] studied the behavior of a small group of spammers, finding that they exhibit very different behavior from non-spammers in terms of posting tweets, replying tweets, followers, and followees. However, they study the behavior of a different form of attack, where users automatically follow a number of other users expecting reciprocity. Similarly, Wang [26] collected thousands users on Twitter and used classification to distinguish the suspicious behaviors from normal ones. In this paper, we focus on spammers that affect search considering a near-complete dataset from Twitter as well as a manually built collection of spammers and non-spammers. More important, we leverage our study about the characteristics of users and propose a spammer detection mechanism.

7. CONCLUSIONS AND FUTURE WORK

In this paper we approached the problem of detecting spammers on Twitter. We crawled the Twitter site to obtain more than 54 million user profiles, all their tweets and links of follower and followees. Based on this dataset and using *manual inspection*, we created a labeled collection with users classified as spammers or non-spammers. We provided a characterization of the users of this labeled collection, bringing to the light several attributes useful to differentiate spammers and non-spammers. We leverage our characterization study towards a spammer detection mechanism. Using a classification technique, we were able to correctly identify a significant fraction of the spammers while incurring in a negligible fraction of misclassification of legitimate users. We also investigate different tradeoffs for our classification approach and the impact of different attribute sets. Our results show that even with different subsets of attributes, our approach is able to detect spammers with high accuracy. We also investigate the feasibility of detecting spam instead of spammers. Although results for this approach showed to be competitive, the spammer classification uses a much larger set of attributes and is more robust to spammers that adapt their spamming strategies.

We envision three directions towards which our work can evolve. First, we intend to explore other refinements to the proposed approach such as the use of different classification methods. Second, we plan to increase and improve our labeled collection in a collaborative manner, including tweets with other popular hashtags. Finally, we aim at investigating other kinds of attacks on Twitter.

Acknowledgments

We sincerely thank Krishna P. Gummadi for his valuable comments and suggestions.

8. REFERENCES

[1] List of spam words. http://codex.wordpress.org/Spam_Words.

[2] Top twitter trends in 2009. http://blog.twitter.com/2009/12/top-twitter-trends-of-2009.html.

[3] Twitter dataset homepage. http://twitter.mpi-sws.org.

[4] Google Adds Live Updates to Results, *The New York Times*, December 2009. http://nyti.ms/cnszI5.

[5] H. Aradhye, G. Myers, and J. Herson. Image analysis for efficient categorization of image-based spam e-mail. In *Int'l Conference on Document Analysis and Recognition (ICDAR)*, 2005.

[6] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, and M. Gonçalves. Detecting spammers and content promoters in online video social networks. In *Int'l ACM Conference on Research and Development in Information Retrieval (SIGIR)*, 2009.

[7] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, M. Gonçalves, and K. Ross. Video pollution on the web. *First Monday*, 15(4), April 2010.

[8] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, and K. Ross. Video interactions in online video social networks. *ACM Transactions on Multimedia Computing, Communications and Applications (TOMCCAP)*, 5(4):1–25, 2009.

[9] P. Calais, D. Pires, D. Guedes, J. W. Meira, C. Hoepers, and K. Steding-Jessen. A campaign-based characterization of spamming strategies. In *Conference on e-mail and anti-spam (CEAS)*, 2008.

[10] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: Web spam detection using the web topology. In *Int'l ACM SIGIR*, 2007.

[11] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi. Measuring User Influence in Twitter: The Million Follower Fallacy. In *Int'l AAAI Conference on Weblogs and Social Media (ICWSM)*.

[12] F. Douglis. On social networking and communication paradigms. *IEEE Internet Computing*, 12, 2008.

[13] R. Fan, P. Chen, and C. Lin. Working set selection using the second order information for training svm. *Journal of Machine Learning Research (JMLR)*, 6, 2005.

[14] D. Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Int'l Workshop on the Web and Databases (WebDB)*, 2004.

[15] S. Garriss, M. Kaminsky, M. Freedman, B. Karp, D. Mazières, and H. Yu. Re: Reliable email. In *USENIX Conference on Networked Systems Design & Implementation (NSDI)*, 2006.

[16] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *Int'l. Conference on Very Large Data Bases (VLDB)*, 2004.

[17] P. Heymann, G. Koutrika, and H. Garcia-Molina. Fighting spam on social web sites: A survey of approaches and future challenges. *IEEE Internet Computing*, 11, 2007.

[18] N. Jindal and B. Liu. Opinion spam and analysis. In *Int'l Conference on Web Search and Web Data Mining (WSDM)*, 2008.

[19] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European Conference on Machine Learning (ECML)*, 1998.

[20] R. Kohavi and F. Provost. Glossary of terms. *Special Issue on Applications of Machine Learning and the Knowledge Discovery Process, Machine Learning*, 30, 1998.

[21] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Int'l World Wide Web Conference (WWW)*, 2010.

[22] Y. Lin, H. Sundaram, Y. Chi, J. Tatemura, and B. Tseng. Detecting splogs via temporal dynamics using self-similarity analysis. *ACM Transactions on the Web (TWeb)*, 2, 2008.

[23] A. Mislove, A. Post, K. Gummadi, and P. Druschel. Ostra: Leverging trust to thwart unwanted communication. In *Symposium on Networked Systems Design and Implementation (NSDI'08)*, 2008.

[24] K. Morik, P. Brockhausen, and T. Joachims. Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring. In *Int'l Conference on Machine Learning (ICML)*, 1999.

[25] A. Thomason. Blog spam: A review. In *Conference on Email and Anti-Spam (CEAS)*, 2007.

[26] A. Wang. Don't follow me: Spam detection in twitter. In *Int'l Conference on Security and Cryptography (SECRYPT)*, 2010.

[27] I. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[28] C. Wu, K. Cheng, Q. Zhu, and Y. Wu. Using visual features for anti-spam filtering. In *IEEE Int'l Conference on Image Processing (ICIP)*, 2005.

[29] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov. Spamming botnets: Signatures and characteristics. In *ACM SIGCOMM*, 2008.

[30] Y. Yang. An evaluation of statistical approaches to text categorization. *Information Retrival*, 1, 1999.

[31] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *Int'l Conference on Machine Learning (ICML)*, 1997.

[32] S. Yardi, D. Romero, G. Schoenebeck, and D. Boyd. Detecting spam in a Twitter network. *First Monday*, 15(1), 2010.