Searching for Spam: Detecting Fraudulent Accounts via Web Search

Marcel Flores and Aleksandar Kuzmanovic

Northwestern University
marcel-flores@u.northwestern.edu, akuzma@cs.northwestern.edu

Abstract. Twitter users are harassed increasingly often by unsolicited messages that waste time and mislead users into clicking nefarious links. While increasingly powerful methods have been designed to detect spam, many depend on complex methods that require training and analyzing message content. While many of these systems are fast, implementing them in real time could present numerous challenges.

Previous work has shown that large portions of spam originate from fraudulent accounts. We therefore propose a system which uses web searches to determine if a given account is fraudulent. The system uses the web searches to measure the online presence of a user and labels accounts with insufficient web presence to likely be fraudulent. Using our system on a collection of actual Twitter messages, we are able to achieve a true positive rate over 74% and a false positive rate below 11%, a detection rate comparable to those achieved by more expensive methods.

Given its ability to operate before an account has produced a single tweet, we propose that our system could be used most effectively by combining it with slower more expensive machine learning methods as a first line of defense, alerting the system of fraudulent accounts before they have an opportunity to inject any spam into the ecosystem.

1 Introduction

As social networks have continued to grow in popularity, so has the problem of spam. The Twitter social network presents a fresh set of challenges to the task of spam detection [1]. The forced brevity of 140 characters has made many of the tools for detecting email spam unusable, as one can no longer depend on legitimate messages being longer [2]. The popularity of URL shorteners further obfuscates messages, making the already difficult task of URL blacklisting even more difficult [1,3]. Social links in the Twitter network are also non-symmetric, complicating detection methods that depend on implicit trust in the network.

While often very effective, current spam detection strategies generally depend on account features that manifest themselves after the account has been active, such as message format and content, as well as position in the social graph. This requirement creates a delay, and even detection methods which are able to train rapidly are unable to stop the first volleys of spam that are injected into the system.

M. Roughan and R. Chang (Eds.) PAM 2013, LNCS 7799, pp. 208–217, 2013.

However, the explosive popularity of Online Social Networks (OSNs) has had another effect: legitimate users often participate in multiple, interlinking, online services. Users will often use the same, or similar names, for various accounts across the web. It is therefore not overly difficult to detect the presence of the same user on multiple sites. In contrast, spammers would have difficulty emulating such a dynamic web presence. While creating fraudulent accounts on a single website may often be possible: creating a batch of coordinated accounts across services would require defeating a varied array of spam detection systems. To make matters worse for spammers, if they create an online persona across services which is flagged as spam in one service, it could be easily linked to its other accounts, making it easier to identify as spam in the remaining services.

Therefore, in order to detect fraudulent accounts, one could measure exactly this distributed online presence. Not only is it extremely robust to any sort of escalation by spammers, it can also be performed quickly and cheaply using existing indices of web content. Through nothing more than a web search, one can measure how frequently an account name, or similar identifier, appears on the web, and therefore determine if the account is likely to be legitimate. Since this check requires only that the user have an account, it does not depend on social graph information or content posted by the user, and can therefore be performed before the user has taken any actions in a particular network.

In this paper, we present a spam detection method which uses the results of web searches for accounts to detect the presence of fraudulent accounts in the Twitter social network. First we consider an overview of the current state-of-the-art methods. We then discuss the in-depth design of our system, and some of the challenges of measuring web presence. Next, we describe an analysis on a collection of actual Twitter accounts, and show that we are able to detect 74.23% of the fraudulent accounts. Finally, we discuss how this procedure could be integrated into existing spam detection workflows and be extended beyond the Twitter network.

2 Background

One common form of spam on Twitter is a "mention," an interaction in which a user uses the name of another user in a message, generating a notification for the user whose name was used. Since the user who performs the mention need not be linked to the receiver in any way, these messages may be unsolicited. Often times these mentions will come in response to the use of a keyword for which lurking spam accounts are watching. For example, the use of the word "phone" in the tweet "I recovered my phone!" by user1, received the reply "@user1 Check out great phone cases! http://nefariouslink.info " from a spam bot with no network links to the original poster. While there are other ways for spam URLs and messages to be distributed through Twitter, this method is both the most disruptive and difficult to avoid.

Previous attempts to detect and measure spam in Twitter and other OSNs have considered a number of information sources. Analysis of the URLs posted by spammers has proven effective in certain cases, and has enabled the categorization of spam messages into larger spam campaigns [4,1]. Another method explicitly analyzes the content of posted URLs and aims to determine if the linked pages are spam [5]. While an important part of spam detection, these techniques often perform too slowly to prevent users from being exposed to spam links [1].

Others techniques use both the content of the messages, profile information, and information from the Twitter social graph to try and determine the nature of tweets [6,7,8,9,10,11]. Even more complex methods have further used similar types of information to determine which large scale campaign a spam tweet belongs [2,4,12,3]. These methods are often effective but rely on complex training and analysis.

We propose the use of near-instantly available outside information to make an initial call on the nature of an account. This system is designed to work alongside existing, more computationally heavy systems that may require significant training time. By combining such systems, one could avoid much of the initial exposure to spam, while still accurately eliminating fraudulent accounts.

Outside information from the web has been used previously in determining context of messages on Twitter [13], however it was largely used for the purposes of classification and analysis of actual tweet text, rather than the detection of spam. While past experiments have suggested that most spam originates from compromised accounts [1], more recent studies have found that this may not be the full picture and that fraudulent accounts contribute significantly to spam on Twitter [3]. Our method therefore focuses on the detection of fraudulent accounts created expressly for distributing spam, rather than a per-message analysis. This decision could allow for accounts to be checked even before they are able to send out any malicious messages, rather than attempting to classify messages as they are sent.

3 Design

Our system is designed to work on the account granularity, and therefore analyzes a given account and attempts to determine if the account is fraudulent. We attempt to perform this determination by measuring a user's web presence beyond Twitter.

There are a number of reasons why one might expect that such information could provide a reasonable means by which to differentiate spam from legitimate users. First, creators and users of spam accounts have incentive to create accounts which are not easily linked to other related entities on the web, as they could then easily be flagged as spam and blacklisted. Furthermore, the cost of creating matching fraudulent accounts on different services would be extremely high, as in general each of these services employ their own spam detection algorithms. Legitimate users, on the other hand, experience the exact opposite incentive: linking a Twitter account to other web services (from forum accounts to blogs to businesses), allows users to reach a larger audience.

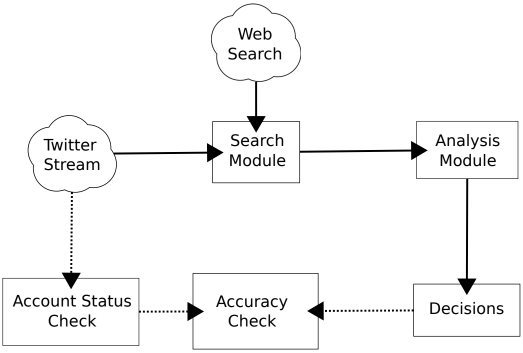

Fig. 1. Spam detection system overview. Dotted lines indicate portions used only in the experiment.

Our system detects these connections, or lack of connections, by use of a web search. Accounts which are easily connected to outside portions of the web are then likely to be legitimate users. We note that this system is not designed to operate on its own as the sole arbiter of spam. Instead, it is designed to act as an additional source of information in a comprehensive spam detection system.

3.1 Methods

We emphasize that our system needs no information from the Twitter network aside from the account's unique username and display name, and can therefore be used on an account as soon as it is created. For our verification in Sect. 4, we only consider accounts which have performed a mention which contains a URL. However this was largely for data collection convenience and is not indicative of any limitation in the system.

Figure 1 provides an overview of the system. First, we feed the input data from Twitter through the search module. This module performs a web search for the username (the unique account name that the user has selected) and the display name (the non-unique name the user has selected for display). We note that Twitter does not require a meaningful display name, and, as a result, many are filled with business names, titles, and nicknames.

After the searches are performed, the result sets are fed into the analysis module. We perform a number of noise-reduction techniques in order to eliminate results that are often returned for any search of a Twitter user, but do not meaningfully distinguish spam and non-spam accounts. We describe these techniques in more detail in Sect. 3.2. Finally, the analysis module examines the remaining results for the account. If there are no results for either the username and the display name, the account is marked as spam. Otherwise, the account is presumed to be legitimate.

3.2 Noise Reduction

The immense popularity of Twitter has resulted in not only many Twitter users, but a number of services designed to add to the Twitter user experience. Many of these services are directed, generating content for all users they find in the network, not just users who actively seek out their service.[1]

Twitter's popularity has also resulted in heavy integration with existing pages. For example, many pages will include a Twitter feed on the page displaying any messages seen relating to the content of the page. Since these services appear in the results for all users, not just legitimate users, we consider them noise. In order to eliminate this noise, we create a blacklist of the domains most commonly returned in search results for all account queries. We then remove the domains on the list from all results the system encounters. In Sect. 4.5 we show that this blacklist can be generated extremely quickly and that a relatively short list is effective in removing noise.

Additionally, it is common that the only result for both a username and a display name is in fact the same page. While this may be the result of a users web activity, we find that these are generally the result of pages which include a Twitter stream that displays both the user's username and display name. Therefore, when a user has a single result for each query which refer to the same page, we remove the matching URLs from both sets.

4 Experiment

4.1 Dataset

Our dataset was collected from the Twitter stream during March 2012. The initial collection contains over 20GB of data collected from a 1% random sample of all Twitter messages. Since this data contain both Twitter control messages and actual user posts, we filter through the set, collecting all messages which contain both a mention (as described in the previous section) and a URL. Since our method relies on the results of a search engine that biases towards results in English, we also eliminate all tweets that are labeled as non-English. Since our analysis is performed on the account level, we remove all messages from accounts that have been seen previously. This filtering leaves us with approximately $110,000$ messages, each corresponding to a unique account. While relatively small, this dataset contains a sufficient number of both fraudulent and legitimate accounts that we are able to observe the effectiveness of the system on real accounts. Both the dataset and the analysis tools have been made available.[2]

4.2 Ground Truth Dataset

In order to measure the performance of our system, we must establish a ground truth of which accounts are spam. Previous work [3] has made use of Twitter's

[1] For example: `http://klout.com`, `http://favstar.fm`, and `http://twittercounter.com`.

[2] `http://eecs.northwestern.edu/~mef294/projects/twitter.html`

current mechanisms by checking the accounts at least 2 weeks after the initial collection and recording which accounts have been suspended. We repeat this procedure here. Additionally, if any accounts were deleted between the initial observation and the later check, we remove them from our set, as there is no way to determine the reason or nature of their removal.

After the two week period, we find that 21.25% of accounts have been suspended, and are therefore, for our experiment, considered fraudulent accounts. It is, however, important to recognize that this includes (1) accounts that were originally legitimate, but were compromised, (2) abusive users who are not necessarily spammers, and (3) genuine fraudulent accounts. We explore the effects of these issues in the next section.

In order to understand the number of spam messages Twitter has missed, we perform a manual inspection of 200 randomly sampled un-suspended accounts. We only mark accounts which are clearly fraudulent as spam. In particular, accounts which started legitimate, but appear to have been compromised later are ignored. In our sample, we find that 36 of the accounts are fraudulent, suggesting that 18% of accounts which Twitter has not suspended are fraudulent. Therefore we suspect that at least some of our false positives will result from this error.

4.3 Performance Measurement

In order to properly measure the performance of our system, we compute its true-positive rate (TPR) and false-positive rate (FPR). The TPR is computed as:

$$TPR = \frac{\#\text{ of true positives}}{\#\text{ of true positives} + \#\text{ of false negatives}}.$$

This tells us what fraction of the spam accounts we were able to correctly identify as spam. The FPR is computed as

$$FPR = \frac{\#\text{ of false positives}}{\#\text{ of false positives} + \#\text{ of true negatives}},$$

which tells us the fraction of messages that we incorrectly marked as spam.

As we noted in Sect. 4.2, our FPR may be inflated by the the presence of spam accounts that have not yet been detected by Twitter. On the other hand our TPR may be underestimating our performance for a number of reasons. First, it is possible that accounts which have been suspended by Twitter are not actually spam, but were suspended for other violations. Second, our system only detects whether or not an account is fraudulent. If the account was once legitimate, i.e. became compromised later, a web search, and therefore our system, will likely indicate that the account is legitimate.

4.4 Results

When properly tuned, our system is able to achieve a TPR of 74.23% at a FPR of 10.67%. While the TPR is similar to those seen with other algorithms [6,2,8,10], direct comparison is difficult due to variations in methodologies. In particular, differences in determining a suitable “ground truth” (Twitter suspension information, URL blacklists, and manual verification) and granularity (account and message levels) mean each study is measuring a slightly different value.

In order to understand how greatly our system is affected by the errors in our ground truth set, we manually classify a random sample of 200 accounts which are marked as being false positives. Again, we only classify an account as fraudulent if it is clear that the account has never performed legitimate tweets. We find that 123, or 61%, of the accounts are clearly fraudulent. Of the remaining 77, 15 had begun tweeting spam URLs after long periods of inactivity. This long period of inactivity likely reduces the visible web presence of accounts, causing our system to flag them as spam. We also note that of the further remaining 62 accounts, an additional 18 are non-English, which we have already indicated our system is not designed to handle. If we consider only those accounts which the Twitter ground truth has missed that were clearly fraudulent, we see that our FPR is potentially as low as 4.5% and the TPR is potentially as high as 79.2%.

4.5 Blacklist Tuning

In order to eliminate much of the noise which results from performing a search for a Twitter name, we generate a blacklist of the 10 most frequently occurring domains for each type of query. These domains are then removed from all lists when performing the analysis. Since we know that they will always constitute noise, we always add the various forms of the Twitter domains to the blacklist (“Twitter.com”, “Twitter.ru”, etc.). Additionally, we perform a reverse DNS on any results which consist of an IP address. If the lookup resolves to an address in the Twitter network, we also add it to the blacklist.

Table 1. A summary of the performance when toggling noise reduction techniques

Method	**TPR**	**FPR**
No Blacklist	62.64	2.61
Blacklist on Display Name	67.30	5.53
Blacklist on Username	70.86	8.22
No Blacklist Exceptions	72.92	9.64
Full	74.23	10.67

In order to prevent the blacklist from eliminating valid sites, we manually select 10 sites which are excluded from blacklist generation. These particular sites were selected as they are among those that appear most often and clearly constitute a web presence. These sites consist of other OSNs (Facebook, LinkedIn, MySpace) and sites with OSN-like features (flickr.com, imdb.com, vimeo.com, soundcloud.com, yelp.com, lockerz.com).

We note that the differences in form of the username and display name result in vastly different results. The username results are often filled with Twitter and other social networking services designed to target account holders. Display name results, on the other hand, are often polluted with directory entries designed to find individuals. We therefore generate separate blacklists, one for each type of search. The performance of the system when each of these techniques is deactivated can be seen in Table 1.

When tuning such a parameter, a natural question that arises, is which length will result in the best performance? In order to test this, we perform the analysis with lengths from 0 to 50 with intervals of size 5, comparing the results. A blacklist length of 0 means that no domains were filtered out, 5 means that the top 5 most common results are removed from each analyzed query, and so on. The results of this analysis can be seen in Fig. 2. As expected, increasing the size of the list lowers the threshold for what is considered spam, increasing both TPR and FPR. However, we note that at 10 sites we achieve the best tradeoff.

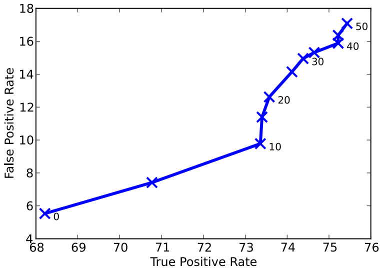

Fig. 2. A blacklist length of 10 seems to provide the best balance between TPR and FPR

Since we would like the list to be generated extremely rapidly so one can obtain meaningful results with minimal delay, we also consider how many sets of results must be considered to produce an effective blacklist. To test this, we generate the blacklist using a random sample of accounts of varying sizes. Figure 3 shows the mean TPR and FPR values for varying sizes of training sets, starting at 5, up to using the entire set of 100, 000. The figure also indicates the standard deviations for each size after 10 iterations. We see that even with a set as small as 500, the quality of the blacklist has already stabilized, as all sets larger than 500 result in similar performance. In an environment such as Twitter, a set of this size could be obtained nearly instantly.

5 Discussion

Given that our method can be applied prior to any activity on the part of the account holder, it would operate best if placed as a first line of defense in

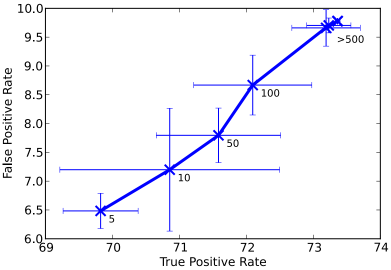

Fig. 3. Both the TPR and FPR stabilize with training sets as small as 500 accounts

spam detection. For example, one could perform our analysis at the time of account creation, using it to inform a more complex system of which accounts are likely fraudulent. Such a system could also be used to place accounts with insufficient web presence on a new-account probation, restricting the amount of spam that such an account could generate before more complex algorithms are able to detect it. Alternatively, users flagged in this manner could be subject to additional verifications in order to obtain full access to their accounts.

Furthermore, we note that our method is by no means limited to Twitter. As it depends only on a broader, more general web presence, the system could be used with any service. In particular, a trend of sites designed to perform single tasks that combine to form a suite of complementary web services (for example Twitter, Tumblr, and Instagram), will likely make web presence easier to detect. In addition, we expect that the growing popularity of cross-logins, which allow users to use the same account to log into multiple sites (popular with Google, Facebook and Twitter accounts), will further aid in detection.

While spammers may attempt to subvert such a system by creating accounts with usernames matching existing accounts on other services, they are still forced to perform a greater amount of manual work for every new account and are potentially limited to a smaller pool of possible accounts.

Additionally, there are natural improvements that could be made to this system to enhance its performance. Rather than considering only the presence of search results to determine if an account is spam, probabilistic methods could be applied. Certain sites that are found to be good indicators could be weighted more heavily, improving the quality of the analysis and further reducing noise.

6 Conclusion

We have presented a system which is able to measure the online presence of a Twitter user by using a web search. By classifying accounts with insufficient presence as spam, we are able to detect 74.67% of fraudulent accounts in a collection of actual Twitter data. Our system is straightforward to implement, and requires no additional content from the suspect accounts, and could therefore be placed as a check at the very beginning of account creation. Furthermore, it has the potential to work extremely well alongside heavier duty algorithms to maximize the amount of spam detected, and minimize spam exposure for legitimate users. Our methods are also generic, and are expected to work equally well beyond Twitter on a number of other web services.

References

1. Grier, C., Thomas, K., Paxson, V., Zhang, M.: @spam: the underground on 140 characters or less. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010, pp. 27–37. ACM, New York (2010)
2. Gao, H., Chen, Y., Lee, K., Palsetia, D., Choudhary, A.: Towards Online Spam Filtering in Social Networks. In: Proceedings of the 19th Annual Network & Distributed System Security Symposium (February 2012)
3. Thomas, K., Grier, C., Song, D., Paxson, V.: Suspended accounts in retrospect: an analysis of twitter spam. In: Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference, IMC 2011, pp. 243–258. ACM, New York (2011)
4. Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y., Zhao, B.: Detecting and characterizing social spam campaigns. In: Proceedings of the 10th Annual Conference on Internet Measurement, IMC 2010, pp. 35–47. ACM, New York (2010)
5. Thomas, K., Grier, C., Ma, J., Vern, P., Song, D.: Design and evaluation of a real-time url spam filtering service. In: 2011 IEEE Symposium on Security and Privacy, SP, pp. 447–462 (May 2011)
6. Benevenuto, F., Magno, G., Rodrigues, T., Almeida, V.: Detecting Spammers on Twitter. In: Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference, CEAS (July 2010)
7. Lee, K., Caverlee, J., Webb, S.: Uncovering social spammers: social honeypots + machine learning. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, pp. 435–442. ACM, New York (2010)
8. Song, J., Lee, S., Kim, J.: Spam Filtering in Twitter Using Sender-Receiver Relationship. In: Sommer, R., Balzarotti, D., Maier, G. (eds.) RAID 2011. LNCS, vol. 6961, pp. 301–317. Springer, Heidelberg (2011)
9. Wang, A.: Don't follow me: Spam detection in twitter. In: Proceedings of the 2010 International Conference on Security and Cryptography, SECRYPT, pp. 1–10 (July 2010)
10. Yang, C., Harkreader, R.C., Gu, G.: Die Free or Live Hard? Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers. In: Sommer, R., Balzarotti, D., Maier, G. (eds.) RAID 2011. LNCS, vol. 6961, pp. 318–337. Springer, Heidelberg (2011)
11. Yardi, C., Romero, D., Schoenebeck, G., Boyd, D.: Detecting spam in a twitter network. First Monday 15(1) (2010)
12. Stringhini, G., Kruegel, C., Vigna, G.: Detecting spammers on social networks. In: Proceedings of the 26th Annual Computer Security Applications Conference, ACSAC 2010, pp. 1–9. ACM, New York (2010)
13. Yerva, S., Miklós, Z., Aberer, K.: What have fruits to do with technology?: the case of orange, blackberry and apple. In: Proceedings of the International Conference on Web Intelligence, Mining and Semantics, WIMS 2011, pp. 48:1–48:10. ACM, New York (2011)