Information Sciences 372 (2016) 332–346

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Twitter turing test: Identifying social machines☆

Abdulrahman Alarifi[a,*], Mansour Alsaleh[a], AbdulMalik Al-Salman[b]

[a] Computer Research Institute, King Abdulaziz City for Science and Technology, P.O. Box 6086, Riyadh 11442, KSA
[b] Computer Science Department, King Saud University, Riyadh, KSA

ARTICLE INFO

Article history:
Received 2 April 2015
Revised 27 July 2016
Accepted 11 August 2016
Available online 20 August 2016

Keywords:
Content spam
Fake user account
Social network
Sybil account
Twitter
Web spam

ABSTRACT

Many machine-controlled Twitter accounts (also called "Sybils") are created each day to provide services, flood out messages for astroturf political campaigns, write fake product reviews, or produce an underground marketplace for purchasing Twitter followers, retweets, or URL advertisements. In addition, fake identities and user accounts in online communities are resources used by adversaries to spread malware, spam, and harmful links over social networks. In social networks, Sybil detectors rely on the assumption that Sybils will find it harder to befriend real users; thus, Sybils that are connected to each other form strongly connected subgraphs, which can be detected using the graph theory. However, a majority of Sybils have actually successfully integrated themselves into real social media user communities (such as Twitter and Facebook). In this study, we compared the current methods used for detecting Sybil accounts. We also explored the detection features of various types of Twitter Sybil accounts in order to build an effective and practical classifier. To evaluate our classifier, we collected and manually labeled a dataset of Twitter accounts, including human users, bots, and hybrids (i.e., tweets posted by both human and bots). We consider that this Twitter Sybils corpus will help researchers to conduct high-quality measurement studies. We also developed a browser plug-in, which utilizes our classifier and warns the user about possible Sybil accounts before accessing or following them after clicking on a Twitter account.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

The Internet is now a fundamental source of information and a vital tool for individuals and businesses, and thus the desire to automatically generate and spread influential content has increased dramatically. This capability can be applied to various objectives, including: (1) providing advertisements; (2) promoting politically oriented views and opinions; (3) promoting financial trends; (4) generating product reviews; (5) spreading malware, spam, and harmful links; (6) influencing search engine results such that particular links are shown first; (7) affecting voting results; (8) generating news feeds; and (9) creating an underground marketplace for purchasing social media followers. Some of these objectives are benign (e.g., news and information accounts), but the majority of these uses can be categorized as fraud. A common and effective method for exploiting this capability is utilizing social media tools via machine-controlled accounts with the hope that these accounts will be perceived as humans.

☆ This manuscript is an extension of an earlier conference version [4].

* Corresponding author. Fax: +966114814553.
E-mail addresses: aarifi@kacst.edu.sa (A. Alarifi), maalsaleh@kacst.edu.sa (M. Alsaleh), salman@ksu.edu.sa (A. Al-Salman).

http://dx.doi.org/10.1016/j.ins.2016.08.036
0020-0255/© 2016 Elsevier Inc. All rights reserved.

At present, billions of people use social networks, such as Facebook, Twitter, LinkedIn, and Google Plus, in their daily lives for different purposes [10,24,34]. Twitter is a social network where users can publish a post as a "micro-blog" or "tweet", but it is limited to 140 characters per tweet. According to the United States Securities and Exchange Commission, the number of monthly Twitter active users exceeds 230 million and over 100 million daily active users generate about 500 million tweets daily [40]. Given these huge volumes, Twitter is a target for spammers and hackers who prey on the less technologically capable. In fact, numerous machine-controlled Twitter accounts (called "Sybils") are created each day to provide services, send spam and harmful links, or to fuel the black market for buying Twitter followers or retweets. We cannot ignore the impact that these accounts have on the online community because they give a false sense of credibility to the poster as well as increasing security and privacy concerns among users of the Web (e.g., because most links posted by a Sybil account tend to be malicious).

In this study, we aimed to mitigate these user concerns by implementing a system to determine whether the user of a Twitter account is a human or a bot. Thus, before following or interacting with a Twitter user, people can know whether the user is a human or a bot, thereby helping to avoid opening tweets or links posted by an account that could be harmful, spam, or inappropriate. However, machine-controlled accounts are constantly evolving to prevent detection such as by randomizing their tweeting times, posting human-like interactions, or avoiding communication with other bots. Therefore, we aimed to find new detection features for detecting bots regardless of the semantic content of the tweets.

The main contributions of this study are as follows.

1. Establishing Twitter Sybils corpus. We collected and manually labeled a dataset of Twitter accounts (including human, bot, and hybrid accounts) to evaluate our proposed classifier. We have also made this dataset available to the public to help other researchers in this area[1].
2. Analyzing detection features. We analyzed various types of Sybil account detection features to obtain an appropriate set of detection features, which facilitate reasonably accurate detection. To the best of our knowledge, we determined several novel features for Twitter Sybil account detection.
3. Twitter Sybils classifier. We built a classifier for detecting Twitter Sybil accounts by using supervised machine learning techniques.
4. Browser detection plug-in. We developed a browser plug-in referred to as the Twitter Sybils Detector (TSD), which utilizes our classifier and warns users about Sybil accounts before accessing them after clicking on a Twitter account[2].

The remainder of this paper is organized as follows. Section 2 describes how we built our Twitter Sybils corpus. Section 3 explains the feature extraction and selection process. In Section 4, we first describe our classifier and we then present an evaluation of its performance based on our dataset. Section 5 considers the system architecture and components. Section 6 presents an overview of related research. In Section 7, we provide our conclusions.

2. Building our corpus

Lack of absolute ground truth (AGT) datasets. The reliable evaluation of anomaly detection systems is challenging owing to the difficulty of validating the detection results and the lack of AGT datasets [5]. An AGT dataset is considered to be an ideal reference, where all of the true positives and true negatives are identified correctly. Some studies have investigated the detection of Sybil accounts on Twitter, but no known dataset can be treated as an AGT. Different approaches can be employed for establishing an AGT, but most of these approaches either misrepresent the sample space or they require a rigorous and costly process. For example, a simulation approach for generating a labeled dataset might not provide a realistic representation of the actual data collected from the source. Another approach is to slowly label a real dataset by using various approaches and rater groups, and to make them publicly available for a period of time for testing and validation by the community.

Deriving a ground truth reference (GTR) dataset. Given the challenges of establishing an AGT dataset and the lack of labeled datasets with an AGT, we derived a GTR dataset (denoted as GTR_1) that uses a reliable labeling mechanism, but it is less rigorous than those usually employed for deriving an AGT. In addition to the development of new dataset features for capturing the up-to-date tactics employed to lure users to Sybil accounts, we consider that our dataset can be used by other researchers as a GTR to evaluate and benchmark potential detection systems. A two-step process was followed to collect the data for GTR_1. First, we randomly collected a large set of tweets by using the Twitter Streaming API to provide random samples of recently published tweets. The collected tweets were then used by the Twitter REST API to gather account information for the users who posted the tweets, which comprised nearly 1.8 million accounts. The account information comprised the user's name, account creation date, account description, profile picture, tweet text, tweet submission time and date, tweet source, number of favorites and retweets for each tweet, number of tweets, number of followees and followers, names and profile pictures of followees and followers, number of times the user had favorited, number of times the user had been favorited, number of lists the user had created, and the number of lists of which the user was a member, which were used later to allow us to differentiate between human, Sybil, or hybrid accounts. Our account was not on Twitter's

[1] Our dataset of Twitter accounts can be downloaded from the following link: https://github.com/TwitterSybilDetector/TwitterSybilDetector

[2] The Chrome browser plug-in can be download from: https://github.com/TwitterSybilDetector/TwitterSybilDetector

Table 1
Details of the datasets.

	Parameter	Value
GTR_2	Source	Lee et al. [31]
	$\|GTR_2\|$	41,499
	$\|GTR_2(Human)\|$	19,276
	$\|GTR_2(Sybil)\|$	22,223
GTR_1	Number of Accounts (p_1 = phase 1)	1.8 million
	Number of Accounts (p_2 = phase 2)	2000
	$\|s \subset GTR_2\|$	1020
	$\|GTR_1 = p_2 \cup s\|$	3020
	$\|p_2(Human)\|$	1390
	$\|p_2(Hybrid)\|$	41
	$\|p_2(Sybil)\|$	569
	$\|s(Human)\|$	520
	$\|s(Sybil)\|$	500
	Number of Features	30
	Gathering Tools and APIs	Twitter REST API & Twitter4j
	Database Engine	MongoDB

whitelist; thus, we had to abide by its 15 minutes limit [44], which was enforced after we called the maximum number of requests by using the REST API. To facilitate the data collection process, we used a Java library called Twitter4j [32]. Next, to manually label and classify the Twitter accounts, we randomly selected 2000 accounts from among the 1.8 million accounts collected in the previous phase.

Volunteer raters and labeling process. The labeling process was divided into three phases: rater selection, rater training, and data labeling. During the first phase, the following criteria were applied for selecting raters : (1) raters should have excellent IT skills in order to understand the concept of a Sybil and its uses; and (2) raters should have active Twitter accounts or be active users of Twitter-related applications and services because it this helped them to understand the Twitter community. GTR_1 was manually labeled by a group of 10 volunteer raters, who comprised both males and females aged between 22–30 years, with B.Sc. degrees in computing science majors, and they were considered domain knowledge experts. Each volunteer was given a maximum of 5 min to label each account; however, on average, they only required a few seconds to label an account. The volunteers could only read and understand English and Arabic content; thus, we ignored Twitter accounts containing content in other languages during the selection of 2000 accounts in phase 2 of GTR_1. It is important to note that we did not rely on any language-dependent classification feature; thus, we could accurately classify Twitter accounts independently of the language used.

During the second phase (i.e., rater training), all 10 raters were given a training session, which started with an introduction to the related concepts and definitions (e.g., a Sybil account on Twitter) as well as descriptions of the various motivations users might have for creating such Twitter accounts. In order to train the raters, we used another previously reported GTR, which employed a different labeling approach [31] (denoted as GTR_2). GTR_2 comprised 22,223 Sybil and 19,276 human Twitter accounts. The data in these accounts were collected from Twitter between December 30, 2009 and August 2, 2010. The account owners may have changed their behavior after the dataset was collected; thus, we avoided updating the content of GTR_2 with new tweets. The raters were exposed to examples of different Twitter accounts, which we extracted from GTR_2. We provided the raters with 50 examples, which comprised equal numbers of Sybil and human accounts. When we explained these examples, we highlighted different Sybil tactics and approaches in order to help them with the rating process. In addition, we explained the labeling procedure and the possible external factors that might negatively affect the labeling accuracy (e.g., distraction) in order to improve the labeling results obtained by the raters.

In order to improve the reliability of the ratings, we measured the labeling ability of the raters and their experience level. First, the raters underwent a series of rating trials to refine their rater procedural skills (e.g., vigilance, rating decision speed, and visual search ability). The raters were observed and feedback was given to them after each group of accounts (10 Sybil accounts and 10 human accounts). The raters were expected to practice on 10 groups, but those who improved and satisfied our accuracy requirement of 95% after the fifth group could quit the training phase and start the next phase of data labeling. We excluded one rater who completed all 10 groups without meeting our accuracy requirement of 95%.

The final phase was data labeling where the raters labeled GTR_1. The number of accounts that needed to be labeled in one day was quite high; thus, we allowed the raters a few days for labeling (i.e., a rater was expected to label 480 accounts a day). In order to simplify the labeling process and to reduce errors, we created a GUI tool for labeling, which showed the account information to the volunteers in an appropriate format to ensure that every account was labeled by all of the volunteers. In addition, the sequence of accounts shown to the raters differed among the raters. The labeling tool was explained to the raters before they started the labeling process, where all of their questions and concerns were answered. The final labeling for every account was derived using majority voting to reduce the effect of human errors. Nearly 69.5% of the accounts obtained were human, 28.5% were Sybils, and 2% were hybrids. Table 1 summarizes the statistics for our dataset and the tools or APIs used. For each rater, we neglected the last 10 labels in any given session because the rater might have been less focused at the end (the neglected labels were included in other sessions for the same user). Foil accounts were

used to evaluate the performance of the raters during the actual labeling process. The accuracy of the majority vote process was 96%. For performance gains, we used the MongoDB database engine[3] to store the dataset collected during all phases.

We used Fleiss' kappa[4] to measure the reliability of the agreement between our nine volunteer raters, which yielded a kappa value of 0.61, thereby indicating substantial agreement between the raters. An earlier labeling attempt without prior training of the raters resulted in a lower kappa value, thereby indicating only fair agreement.

Using foils to evaluate the accuracy of the raters. In order to evaluate the accuracy of labeling for our dataset GTR_1, we randomly selected a subset of 1020 Twitter accounts from GTR_2 and injected them into GTR_1 in order to obtain a rough estimate of the accuracy and reliability of our labeling process. Thus, to estimate the accuracy and reliability of our domain knowledge experts in labeling the corpus, we used 1020 previously labeled Twitter account foils (the raters were blind to the nature of the foils), which increased the size of GTR_1 to 3020 accounts. More detailed descriptions of both GTR_1 and GTR_2 are presented in Table 1. A sample size of 1020 from a total population size of 3020 means that the margin of error[5] was less than $\pm$ 2.5% with a confidence level of 95%. The results obtained after our nine volunteers labeled these 1020 previously labeled Twitter account foils (using majority voting) showed that their accuracy was 96%, i.e., we needed to consider this 4% error rate during the labeling process to determine the accuracy of the proposed classifier. It should be noted that while such an effect on the classifier could slightly reduce the accuracy rate by increasing false alarm or false negative rates, it could also increase the detection accuracy because some of the false alarms might actually have been positive samples, which were mistakenly flagged as negative by the human raters.

3. Feature extraction and selection

The feature extraction and selection process is a critical preliminary step. If the input is too large to be processed and there might be some redundancy, then the input needs to be transformed into a reduced representation set of features. Various features have been proposed previously for maximizing the inter-class variability while minimizing the intra-class (e.g., [8,16,18,37,41]. Using the actual raw data for classification may yield a complex classifier structure as well as causing difficulty in the training phase, thereby resulting in poor classification performance. Thus, we propose some novel features (to the best of our knowledge) as well as using some previously described features. Table 2 explains our proposed features. For most of the features, we computed the average and standard deviation as inputs for the classifier (the statistical measures for these features are shown in Table A.9 in Appendix A). We were inspired by previously described Twitter spam detection methods, which extract many of these features (the novel features for Sybil detection are denoted by stars in Table 2). It is important to note that we rely completely on features that can be extracted from the Document Object Model (DOM) [6] content of the retrieved Twitter accounts in the browser, rather than requesting further information from Twitter via its APIs, accessing and evaluating the content of the hyperlinks posted within the tweets (e.g., identifying malicious links), or using external tools to check additional properties of the account.

In order to speed up our classifier, shorten the training time, enhance generalization, and reduce overfitting, we applied feature selection methods to select a subset of relevant features for our classification models with the best performance, thereby avoiding features that provide no further information for the classifier. We used and compared the results obtained by two feature selection methods (see Table 3): a correlation based method [23] and a principal components analysis method [26]. The former method favors the features that are correlated with the classes rather than with each other. This method can identify effective features that provide value for the classification process. Principal components analysis is a statistical method for reducing a set of features into a smaller set of linearly uncorrelated variables. However, this method lacks the interpretability of the original features owing to the linear relationship with the selected features [28]. We applied both methods to our dataset. Summaries of the settings for these two feature selection methods as well as the evaluation results obtained are shown in Table 3. Given that the results obtained by the first method (eight selected features) comprised a subset of the 11 features selected by the second method, we used the features from the first method in the training phase for the classifiers (as shown in Table 4).

We examined each of the selected eight features by analyzing the cumulative distribution function for each of them, as shown in Fig. 1. For the first feature, the Sybil accounts (i.e., machine-controlled accounts) had the highest average number of characters per tweet, which might indicate that automated tweets tend to be longer than others, possibly because they include links and other advertising material. As expected, the second feature showed that the number of characters per tweet was convergent for Sybil accounts compared with real accounts. Clearly, Sybil accounts had a higher number of hashtags per tweet compared with regular real accounts (see F3 in Fig. 1). According to feature four, real Twitter users tended to have a higher number of mentions per tweet compared with machine-controlled accounts, possibly owing to the lack of

[3] A cross-platform document-oriented NoSQL database engine.

[4] Fleiss' kappa is a statistical measure for evaluating the reliability of the agreement between a number of raters when assigning categorical ratings to a number of entries [29].

[5] The margin of error (or confidence interval) indicates how close the result for a specific sample size is relative to the true value for the population. Thus, it indicates our uncertainty regarding an unknown parameter, where a very wide interval may indicate that more data should be collected before we can obtain a definite decision.

[6] The Document Object Model (DOM) is a platform and language-independent interface, which defines the logical structure of objects in some document types, such as HML, XHTML, and XML, thereby allowing dynamic access and content updating.

Table 2
Descriptions of features.

Feature name	Desciption
Number of hashtags per tweet	When a user posts a tweet with more than one hashtag, the tweet reaches a wider audience. Thus, bots use multiple hashtags to spread links to the tweets in order to reach a bigger demographic.
*Number of times a hashtag has been used**	Users tweet with the same hashtag multiple times to continue a train of thought or to support the cause of the hashtag. However, bots may use it to flood the hashtag with tweets while the hashtag is trending.
Number of links	Posting links with many tweets could indicate the behavior of a bot because bots are mostly used to spread links to various websites to serve the purposes of bot owners.
Number of mentions per tweet	Mentions are tweets that are directed to users when their names appear in a tweet. If users receive more tweets, the author has less space to write a message, which could mean that the user's goal was not to send an informative tweet, where this behavior is typical of a bot.
Number of characters per tweet	Some bot accounts use fewer words, especially when posting links.
*Number of pictures**	Posting excessively high or low numbers of pictures could indicate bot behavior.
*Tweeting two or more tweets at the exact same time**	Sending two or more tweets at the exact same time is considered impossible for humans, even with copy and paste.
*Profile picture containing a face**	A profile picture containing a face may indicate a personal account, which is not likely to be a bot account.
*Mentioning different users in the same text**	Sending two or more mentions with the same text to different users could mean that the sender is trying to spread links, or that the mention is automated. (Not implemented)
Ratio of followers to number followed	If a user follows a much number greater than the number followed, this might be because his account does not add anything to their feed, which is a sign of bot accounts. The formula for calculating the followers to followed ratio is [14]: $\text{Ratio} = \frac{\text{Number of followers}}{\text{number of followers}+\text{number followed}}$
*Number of times the user has been retweeted**	If a user is not retweeted by others, then this is because they do not find any informative or interesting content, which is typical of bot accounts.
*Number of times the user has favorited**	Favorites indicate interactions and acknowledgments between users, which are a sign of human behavior.
*Number of times the user has been favorited**	Favorites by others indicate interesting and informative content, which is typical of humans.
*Number of favorites compared to the number of tweets**	Tweeting many times with few or no favorites could mean that the user does not read the tweets within the timeline, which is a bot behavior. (Not implemented)
*Number of lists where the user is listed**	Being added to lists indicates that the user is active and they post interesting tweets.
*Whether the user has lists or not**	Possessing lists indicates interactivity between the user and their peers.
*Whether the user employs the basic twitter profile picture**	The default twitter profile picture is employed mostly by new users who are not active, or bots.

Table 3
Feature selection results.

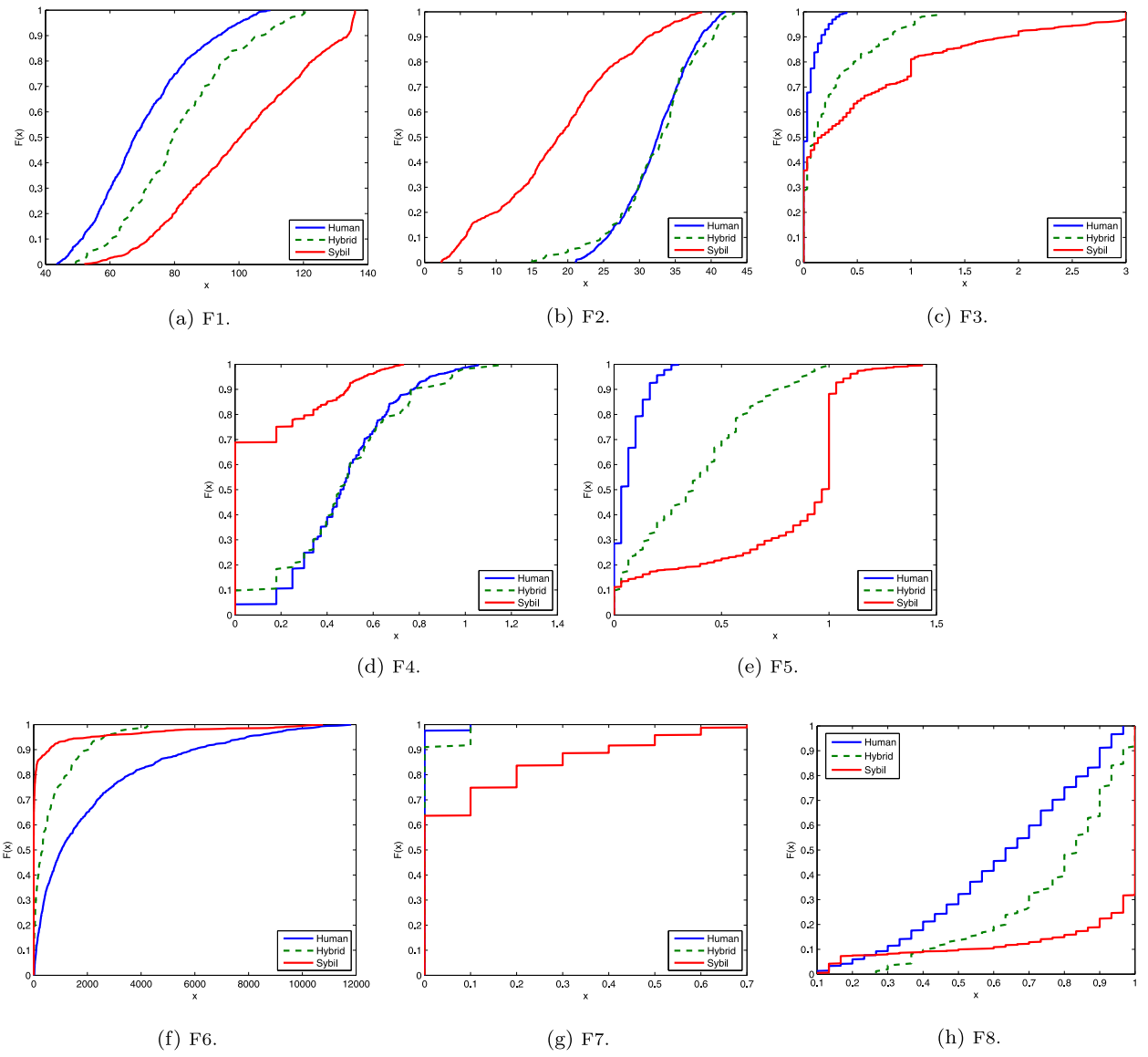
Attribute evaluator	Search method	Number of selected features
CfsSubsetEval: Evaluates the value of a subset of attributes by considering the individual predictive ability of each feature as well as the degree of redundancy between them.	GreedyStepwise: Performs a greedy forward or backward search through the space of attribute subsets.	8
PrincipalComponents: Performs principal components analysis and transforms the data.	Ranker: Ranks attributes based on their individual evaluations.	11

interaction with real users. Sybil accounts had a much higher average number of links than real users, which were probably due to links to advertisements. Another distinctive feature is the number of times a user had been favorited, where number was higher for real users. In terms of the average number of tweets generated at the exact same time, virtually none of the real Twitter users posted tweets at the same time. Surprisingly, for the average number of times the user was retweeted, we found that Sybil tweets were retweeted slightly more than those of real users.

We calculated the probability density functions for all combinations of features. Fig. 2 shows a representative subset where delta is significant between the human and Sybil accounts. For example, in Fig. 2a (the X axis represents feature $F2$ and the Y axis represents feature $F5$; the human class is in red and the Sybil class is in green), we note that there is an obvious peak where the probability density function for the human class is much greater than that for the Sybil class, which is less than 0.5, and there are two other smaller peaks for the opposite case. Fig. 2b shows the delta value (i.e., $|P_h - P_s|(F2, F5)$). Similarly, in Fig. 2c, we note that there is one obvious high peak where the probability density function for the human class is greater than that for the Sybil class, and two obvious lower peak for the opposite case. Fig. 2d shows the delta value (i.e., $|P_h - P_s|(F1, F5)$). Similar observations hold for the other diagrams (i.e., Fig. 2e and Fig. 2g).

Table 4
Features selected (see Table 2 for descriptions).

Feature no.	Feature name
F1	Average number of characters per tweet
F2	Standard deviation of the number of characters per tweet
F3	Average number of hashtags per tweet
F4	Number of mentions per tweet
F5	Average number of links
F6	Number of times the user has been favorited
F7	Average number of tweets generated at the exact same time
F8	Average number of times the user has been retweeted



(a) F1. (b) F2. (c) F3.

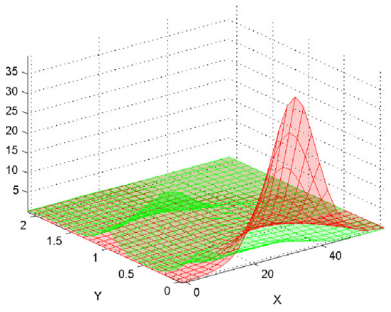
(d) F4. (e) F5.

(f) F6. (g) F7. (h) F8.

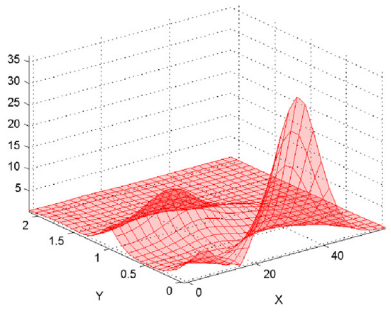
Fig. 1. Cumulative distribution functions of different features.

4. Classification and evaluation

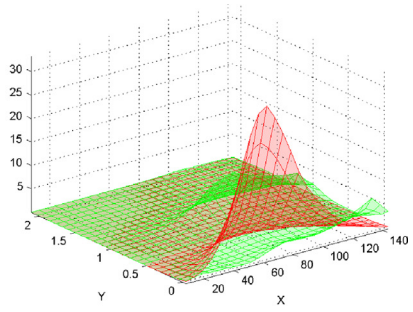
We trained and tested our classification models by using four machine learning algorithms: *decision tree*, *Bayesian network*, *support vector machine (SVM)*, and *multilayer artificial neural network*. A decision tree is a statistical-based algorithm, which selects attributes at the tree nodes starting from the root to the leaves of the tree and the data are divided into sub-



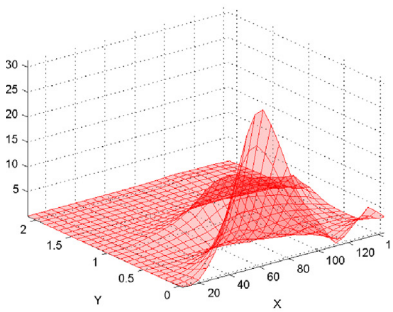
(a) $P_h(x, y)$ and $P_s(x, y)$, where x =F2 and y =F5.



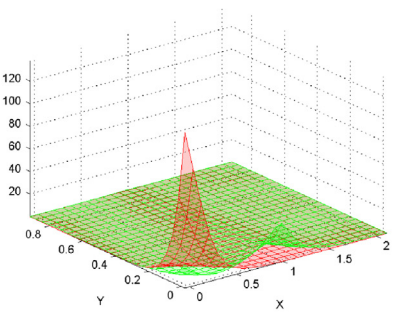
(b) $|P_h - P_s|(x, y)$, where x =F2 and y =F5. $\frac{|P_h - P_s|}{P_h + P_s}$ =0.8052.



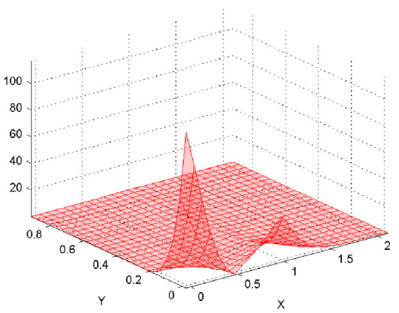
(c) $P_h(x, y)$ and $P_s(x, y)$, where x =F1 and y =F5.



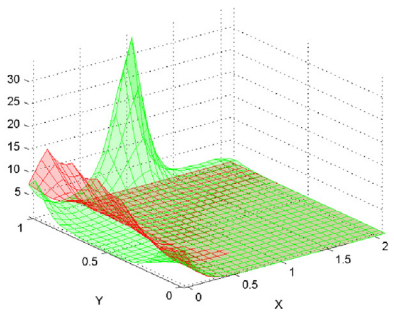
(d) $|P_h - P_s|(x, y)$, where x =F1 and y =F5. $\frac{|P_h - P_s|}{P_h + P_s}$ =0.7896.



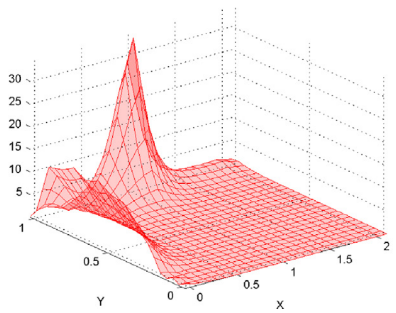
(e) $P_h(x, y)$ and $P_s(x, y)$, where x =F5 and y =F7.



(f) $|P_h - P_s|(x, y)$, where x =F5 and y =F7. $\frac{|P_h - P_s|}{P_h + P_s}$ =0.7452.



(g) $P_h(x, y)$ and $P_s(x, y)$, where x =F5 and y =F8.



(h) $|P_h - P_s|(x, y)$, where x =F5 and y =F8. $\frac{|P_h - P_s|}{P_h + P_s}$ =0.7251.

Fig. 2. Probability density functions P_h and P_s for different combinations of features, where h denotes human accounts (in red) and s denotes Sybil accounts (in green). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sets at each node according to the values of the attributes. Various algorithms can be used to build decision trees such as C4.5 and random forest. The C4.5 algorithm generates a single decision tree for classification whereas random forest tends to create multiple trees for classification. We tested four types of decision trees: J48 (C4.5), logistic model tree, random forest, and logitboost (see Table B.10 in Appendix B). A Bayesian network is a probabilistic representation of the relationships between the features considered using graph theory. SVM is another statistical-based algorithm, which builds a classification model from a set of training examples[7]. We tested two types of SVM: SMO-P and SMO-R (see Table B.10 in Appendix B). A multilayer artificial neural network is a mathematical model, which comprises simple interconnected processing units (neurons) and their associated weights for classification (we used two types, as shown in Table B.10 in Appendix B).

[7] SVMs separate data classes with a set of hyperplanes, but for non-linearly separable data, kernel functions are used to transform the original data into linearly separable data.

Table 5
Accuracy of two-class classification.

Algorithm	Performance measurement indices											
	DR	ER	TP		FP		Precision		Recall		F-measure	
			Human	Sybil	Human	Sybil	Human	Sybil	Human	Sybil	Human	Sybil
J48	89.33	10.67	0.87	0.92	0.08	0.13	0.91	0.88	0.87	0.92	0.89	0.90
			0.89		0.11		0.89		0.89		0.89	
LMT	90.10	9.90	0.95	0.79	0.21	0.05	0.92	0.86	0.95	0.79	0.93	0.82
			0.90		0.17		0.90		0.90		0.90	
RFT	91.39	8.61	0.91	0.92	0.08	0.09	0.92	0.91	0.91	0.92	0.91	0.92
			0.91		0.09		0.91		0.91		0.91	
LBT	89.48	10.52	0.94	0.79	0.21	0.06	0.92	0.84	0.94	0.79	0.93	0.81
			0.90		0.17		0.89		0.90		0.89	
BayesNet	90.90	9.10	0.92	0.90	0.10	0.08	0.90	0.92	0.92	0.90	0.91	0.91
			0.91		0.09		0.91		0.91		0.91	
SMO-P	90.30	9.70	0.94	0.82	0.19	0.06	0.93	0.85	0.94	0.82	0.93	0.83
			0.90		0.15		0.90		0.90		0.90	
SMO-R	80.81	19.19	0.99	0.35	0.65	0.01	0.79	0.96	0.99	0.35	0.88	0.52
			0.81		0.46		0.84		0.81		0.78	
MLP-GD10	89.54	10.46	0.93	0.82	0.18	0.07	0.93	0.82	0.93	0.82	0.93	0.82
			0.90		0.15		0.90		0.90		0.90	
MLP-GD20	89.43	10.57	0.93	0.82	0.18	0.07	0.93	0.82	0.93	0.82	0.93	0.82
			0.89		0.15		0.89		0.89		0.89	

During the training phases, the weights were adjusted using a training algorithm such that the network could classify new examples and obtain the required knowledge. Table B.10 in Appendix B shows the parameter setting for the four algorithms. For all of the classifiers, we performed tenfold cross-validation (leave-one-out) to remove a subset of the observations in turn, before constructing the classifier, and then determining whether this leave-one-out classifier correctly classified the removed observation(s). To overcome overfitting in the decision tree classifier, we used a pruning technique to reduce the size of the decision tree by removing less significant nodes from the tree when classifying instances.[8] A validation threshold was used for the other classifiers to stop training as soon as the algorithms detected overfitting and increased misclassification of the validation set.

We considered two scenarios to train our classification models: (i) two-class classification and (ii) three-class classification, where (i) assumed that we only had two categories of Twitter accounts, i.e., human and Sybil accounts (hybrid accounts are neglected in this case), whereas (ii) assumed that we had three categories, i.e., human, hybrid, and Sybil. In order to handle unbalanced data in the three-class scenarios, we used a Synthetic Minority Oversampling TEchnique (SMOTE)[9] filter to resample a dataset by applying synthetic minority oversampling techniques with an increased percentage of multiple of magnitudes for the number of samples in the hybrid category. For each scenario, we used the following performance measurement indices to compare the performance of the classifiers.

- **Detection rate (DR)** Ratio of the number of correctly classified instances relative to the total number of instances.
- **Error rate (ER)** Ratio of the number of incorrectly classified instances relative to the total number of instances.
- **True positive (TP) for class** *x* Ratio of the number of instances classified correctly as class *x* relative to the total number of instances.
- **False positive (FP) for class** *x* Ratio of the number of instances classified incorrectly as class *x* relative to the total number of instances.
- **Precision for class** *x* Ratio of the number of instances classified correctly as class *x* relative to the total number of instances in class *x*.
- **Recall for class** *x* Ratio of the number of instances classified correctly as class *x* relative to the total number of correctly classified instances.
- **F-measure for class** *x* The harmonic mean of precision and recall for class *x*.

The classification performance results for both scenarios are shown in Tables 5 and 6. Table 5 shows that the RFT and Bayesian network (BayesNet) classifiers (see Table B.10 in Appendix B for the abbreviations of the algorithms) performed slightly better than the other classifiers at two-class classification. Similarly, for three-class classification, Table 6 shows that the RFT classifier obtained the best DR and lowest ER among the other classifiers. The BayesNet classifier had the next best performance, with the second highest DR. The worst classifier in terms of both detection and ER was SMO-R, but we note that the results obtained by all of the classifiers did not differ greatly from each other.

[8] A pruning technique was also useful for reducing the complexity of the final classifier, which reduced the time required to run the final classifier with the browser plug-in.

[9] Synthetic Minority Oversampling TEchnique (SMOTE) is a technique that overcomes the problem of imbalanced dataset by over-sampling the minority by using "synthetic" examples rather than by over-sampling with replacement.

Table 6
Accuracy of three-class classification.

Algorithm	Performance measurement indices																	
	DR	ER	TP			FP			Precision			Recall			F-measure			
			Human	Hybrid	Sybil	Human	Hybrid	Sybil	Human	Hybrid	Sybil	Human	Hybrid	Sybil	Human	Hybrid	Sybil	
J48	83.73	16.27	0.90	0.81	0.73	0.14	0.06	0.07	0.88	0.80	0.75	0.90	0.81	0.73	0.89	0.80	0.74	
				0.84			0.11			0.84			0.84			0.84		
LMT	82.35	17.65	0.89	0.77	0.71	0.18	0.06	0.06	0.86	0.78	0.77	0.89	0.77	0.71	0.88	0.78	0.74	
				0.82			0.13			0.82			0.82			0.82		
RFT	88.00	12.00	0.93	0.91	0.74	0.14	0.03	0.04	0.89	0.89	0.84	0.93	0.91	0.74	0.91	0.90	0.78	
				0.88			0.09			0.88			0.88			0.88		
LBT	81.56	18.44	0.91	0.71	0.68	0.21	0.07	0.05	0.84	0.75	0.80	0.91	0.71	0.68	0.88	0.73	0.74	
				0.82			0.14			0.81			0.82			0.81		
BayesNet	86.74	13.26	0.92	0.90	0.90	0.16	0.02	0.06	0.87	0.95	0.77	0.92	0.90	0.72	0.89	0.92	0.74	
				0.87			0.11			0.87			0.87			0.87		
SMO-P	73.39	26.61	0.94	0.47	0.49	0.35	0.10	0.04	0.76	0.59	0.77	0.94	0.47	0.49	0.84	0.53	0.60	
				0.73			0.23			0.73			0.73			0.72		
SMO-R	61.23	38.77	1	0	0.30	0.85	0	0.01	0.59	0	0.94	1	0	0.30	0.74	0	0.45	
				0.61			0.47			0.53			0.61			0.51		
MLP-GD10	80.14	19.86	0.91	0.65	0.69	0.20	0.08	0.06	0.85	0.71	0.77	0.91	0.65	0.69	0.88	0.68	0.73	
				0.80			0.14			0.80			0.80			0.80		
MLP-GD20	80.85	19.15	0.91	0.66	0.72	0.19	0.08	0.06	0.85	0.72	0.78	0.91	0.66	0.72	0.88	0.69	0.75	
				0.81			0.14			0.81			0.81			0.81		

Table 7
Confusion matrix for two-class classifiers.

Classes		Human		Sybil	
Classified as		Human	Sybil	Human	Sybil
Algorithms	J48	1286	104	127	442
	LMT	1317	73	121	448
	RFT	1307	83	115	454
	LBT	1302	88	118	451
	BayesNet	1300	90	126	443
	SMO-P	1305	85	105	464
	SMO-R	1382	8	368	201
	MLP-GD10	1287	103	102	467
	MLP-GD20	1287	103	104	465
Number of samples		1390		569	

Table 8
Confusion matrix for three-class classifiers.

Classes		Human			Hybrid			Sybil		
Classified as		Human	Hybrid	Sybil	Human	Hybrid	Sybil	Human	Hybrid	Sybil
Algorithms	J48	1244	56	90	66	462	46	97	57	415
	LMT	1239	66	85	95	443	36	108	57	404
	RFT	1289	29	72	43	522	9	115	36	418
	LBT	1270	59	61	131	410	33	104	79	386
	BayesNet	1275	12	103	43	515	16	144	18	407
	SMO-P	1310	67	13	232	271	71	172	119	278
	SMO-R	1383	0	7	570	0	4	401	0	168
	MLP-GD10	1260	70	60	138	375	61	87	87	395
	MLP-GD20	1261	69	60	137	378	59	81	79	409
Number of samples			1390			574			569	

In general, we found that the RFT and BayesNet classifiers performed better than the other algorithms at both two-class and three-class classification. In addition, all of the classifiers could classify human and Sybil accounts better than hybrid accounts, which was due partly to the mixed behavior of the human and bot tweet posts in hybrid accounts. Therefore, the accuracy of two-class classification was higher than that of three-class classification. These observations are also supported by the confusion matrices for the two-class and three-class classifiers in Tables 7 and 8, respectively.

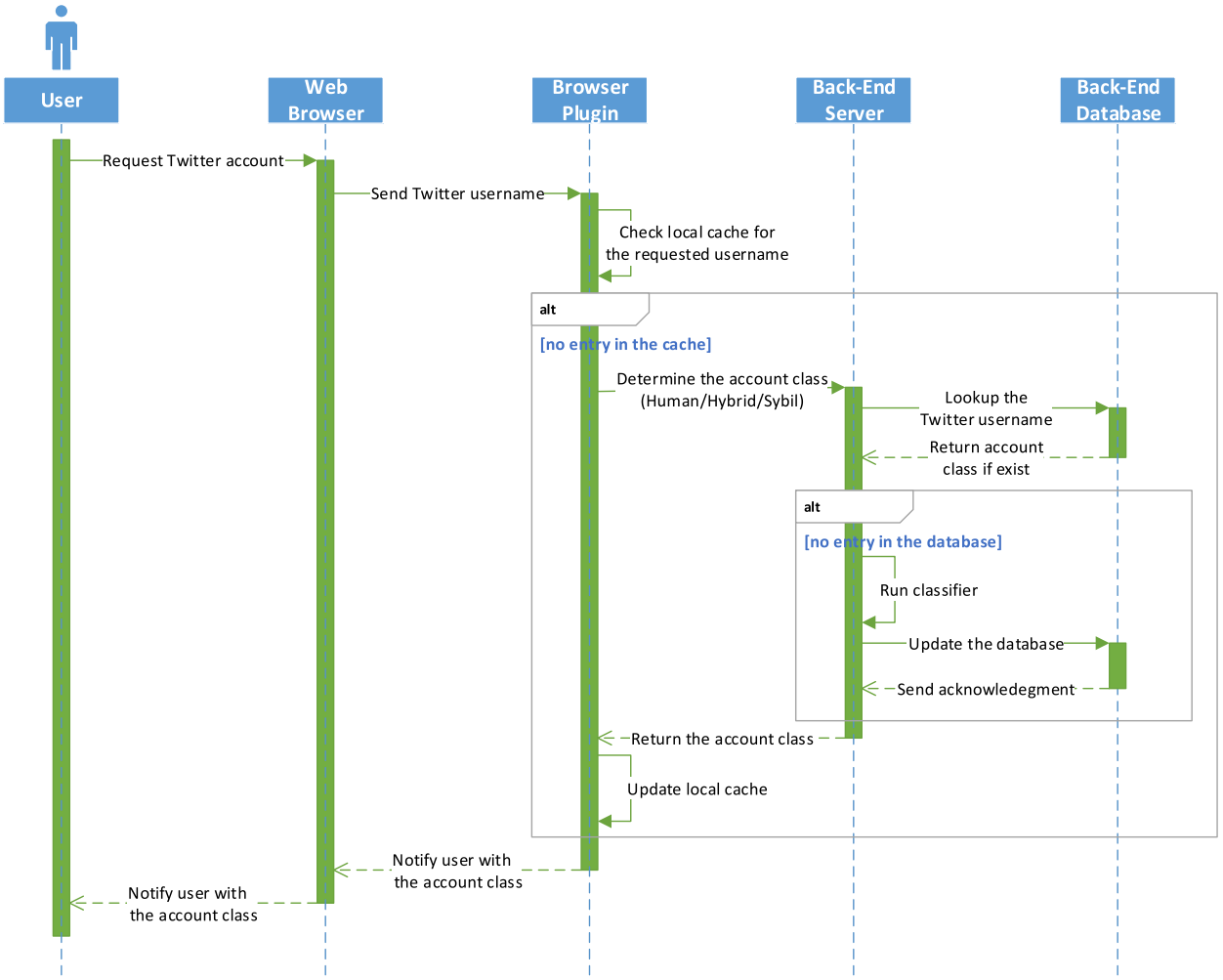


Fig. 3. Sequence diagram showing the relationships between the processes in our system.

5. System architecture and design

Our proposed system comprises two main components: (i) a web browser plug-in and (ii) a back-end server. The web browser plug-in serves as an interface between the browser and the back-end server (see Fig. 3). The plug-in captures the Twitter user name, which is either clicked on by the user or entered by the user via the browser address bar. The plug-in sends the Twitter user name to the back-end server, which extracts the account information and detection features, and then classifies the account as human, hybrid, or Sybil (i.e., these features are used as inputs by the classifier built during the training phase).

The plug-in warns the user if the Twitter account is flagged as a Sybil account; thus, the user can take an appropriate action; otherwise, it will show the user a green color to indicate that the account is human or a blue color to indicate that the account is hybrid. The browser plug-in updates its cache (including Sybil, human, and hybrid account lists) such that only the new Twitter accounts are sent to the back-end server. The back-end server also maintains a database containing the results of all the previous requests received from the clients' plug-ins. This database serves as a local cache lookup mechanism to speed up the process and to collect feedback, which can help to improve the system.

6. Related work

Social bots increasingly use advanced techniques to search for information online; thus, they initiate conversations in order to enter into a dialog and create substantive relationship with real twitter users. This then helps to capture more followers with a greater influence [6,19,25]. In order to evade detection tools, some social bots emulate human temporal signatures, thereby making the bots resemble legitimate accounts when inspected by detection tools [20].

Recently, several detection methods have been proposed that focus on detecting machine-controlled accounts in social networks. Viswanath et al. studied different Sybil defense schemes and showed that most schemes rely on account connectivity, where accounts connected with more trusted accounts have a higher chance of being legitimate [45]. Relying on connectivity allows researchers to employ existing algorithms in real-world environments to detect Sybil accounts in social networks. However, this approach also has limitations because in many social networks, legitimate accounts form multiple connected communities rather than a single one, which makes it harder to distinguish between legitimate and Sybil accounts. SybilRank is another social graph-based solution for Sybil detection by building a social graph to distinguish different accounts, where this tool uses a set of known legitimate accounts (trust seeds) [12]. Similarly, SumUp, Gatekeeper, and SybilLimit are also social graph-based solutions, which consider how Sybil accounts are connected with others [42,43,52]. However, this feature can be mimicked by advanced social bot developers to make their accounts appear similar to real ones [6].

Instead of relying on the links between accounts, conversations can be examined because legitimate accounts tend to have more thorough conversations with other legitimate accounts compared with Sybils [36,50]. Thus, user meta-data have been used as predictive features in other studies [25,49]. Thus, Wagner et al. studied data from the Social Bot Challenge to develop predictive models by using network, behavioral, and linguistic features to identify susceptible accounts, which could be targeted and be infected by social bots, rather than detecting the social bots themselves [46]. This dataset was based on data collected from the interactions of three newly developed social bots on Twitter as part of a challenge to influence user behavior.

Using Renren, a large Chinese social network, and LinkedIn, Wang et al. studied the clickstreams generated by 16,000 real and Sybil user HTTP requests [48], where they analyzed the clickstreams and computed the similarity between them to classify behavioral clusters. They then constructed a Sybil detection system by adding a cluster-less clickstream to the nearest cluster, where if the nearest cluster is a Sybil cluster, then the user is a Sybil. Chu et al. examined over 500 K Twitter accounts to determine the similarities and differences between human, bot, and cyborg users [14]. They created a system based on four components: (1) an entropy-based component; (2) a spam detection component that analyzes a tweet's content to identify spam; (3) an account properties component (e.g., tweeting device to determine whether the account is verified or not); and (4) a decision maker. The decision maker uses the features provided by these three components to determine whether an unclassified user is a human, bot, or cyborg.

Moreover, Messias et al. developed a new method for investigating the susceptibility of Twitter bot account based on widely used influence scores (such as Klout and Twitalyzer tools) for intentional manipulation [33]. The results showed that the influence scores are vulnerable to simple manipulation attacks and thus it is necessary to perform a significant review to deal with social bots. Boshmaf et al. studied the influence of large-scale organized campaigns by social bots on the behavior of Facebook users, which could result in privacy breaches [11]. The results showed that Facebook users are highly vulnerable to this type of attack and some OSN security defenses are practically ineffective for preventing them. Another study examined the characteristics and strategies employed by Twitter bots to increase their chance of success [19]. In order to evaluate the effectiveness of Twitter bot strategies, a 2^k factorial design experiment was used to test 120 Twitter bot accounts with different behaviors.

Kim et al. and Lee et al. proposed the use of geo-tagged tweets to detect Twitter bots rather than temporal information [27,30], where they claimed a high rate of accuracy, but the availability of geolocation information is very limited. Radziwill and Benton used Monte Carlo simulations to explore the structure of time maps to analyze the inter-arrival patterns of tweets in Twitter accounts and identify bot accounts [38].

Truthy employed content-based features and machine learning algorithms, where visualizations were used to detect Twitter bots during U.S. midterm elections [39]. Crowdsourcing was used for data diffusion and flagging was later used for training the system. Wang et al. studied the feasibility and accuracy of crowdsourcing for detecting Sybil accounts because accounts can be classified more easily by humans than current computer algorithms [49]. Other studies employed synchronization information among social bots to determine when the bots orchestrate their activities among themselves in order to allow their detection [10,13,48,51].

Hybrid accounts (cyborgs) are even harder to detect because they behave in a similar manner to both legitimate and bot accounts [53]. Alvisi et al. used multiple approaches to complement each other and obtain better accuracy [6]. Similarly, Wang et al. and Yang et al. combined behavioral information and timing information to facilitate Sybil detection [48,51].

Similarly, several studies have been performed to detect spammers in social networks. Spamming has been studied intensively in different areas such as email and web spamming; however, social network spamming is still an active area of research and development [1,2,7,15,17,21,22,35]. Several spam detection systems have been proposed because spammers are becoming more creative in terms of how they approach spamming. Wang collected a dataset comprising about 25 K users, 500 K tweets, and 49 M followers [47], which was used to examine the benefits of different classification algorithms such as decision tree, artificial neural network, SVM, and naive Bayesian methods. Experiments showed that the naive Bayesian classification algorithm achieved an accuracy of 89%, which was better than the other algorithms.

Benevenuto et al. collected a dataset comprising nearly 54 M user accounts, 1.8 B tweets, and almost 1.9 B social links (they ignored nearly 8% of the accounts, which were set as private) [9]. They then applied selection criteria based on some desired properties, such as having a significant number of accounts in each class, while maintaining the randomness of

account selection in order to label about 1.5 M user accounts containing 4 M tweets. They defined two types of attributes to distinguish between spammers and non-spammers: (1) content attributes (e.g., the fraction of tweets containing links) and (2) user behavior attributes (e.g., the number of followers per number of followees), before employing SVMs to classify a user as a spammer or non-spammer. The system could successfully classify about 70% of the spammers and 96% of the non-spammers, while highlighting the most important attributes used to detect spam on Twitter. In a recent study, Almaatouq et al. presented a unique analysis of the behavioral characteristics of spammers on Twitter [3], where they demonstrated the existence of two behaviorally distinct categories of spammers.

7. Summary and future directions

There has been a dramatic increase in the automatic generation and spread of influential content on social networks such as Twitter, which can be exploited to meet various objectives (e.g., promoting politically oriented views, or spreading malware, spam, and harmful links). Effective methods can be employed to exploit social media tools by using machine-controlled accounts such that these accounts are perceived as humans. In this study, we presented the TSD system, which utilizes supervised machine learning techniques to dynamically detect Twitter Sybil accounts and warn users before accessing or following these accounts. We also built a labeled Twitter Sybils corpus, which we tested with our classifier, and we consider that other researchers could use this corpus to develop and evaluate similar systems. We are also collecting and labeling a larger dataset, and we consider that this larger Twitter Sybils corpus will help researchers to conduct reliable measurement studies. Our classification results showed that the DR was satisfactory for this particular application. Possible future research directions include examining further new key features to enhance the accuracy of detecting Twitter Sybils, identifying possible evasion mechanisms, and exploring the applicability of the classifier to other similar applications.

Acknowledgments

We thank the anonymous reviewers for their comments which helped improve this paper to its present form. This work was supported in part by KACST.

Appendix A. Statistical measures for features

The statistical measures for our selected features are shown in Table A.9.

Table A.9
Statistical measures for each feature (MAD: mean absolute deviation, SD: standard deviation).

		Feature number							
		1	2	3	4	5	6	7	8
Human	**Mean**	70.42	33.41	0.54	0.10	0.03	2598.50	0.01	0.65
	MAD	14.16	4.11	0.20	0.08	0.03	2794.77	0.01	0.18
	Median	67.68	33.46	0.52	0.06	0.01	905.00	0	0.68
	Mode	45.91	35.26	0	0.01	0.01	0	0	0.65
	SD	17.78	5.41	0.27	0.13	0.09	5111.86	0.03	0.22
	Kurtosis	2.98	7.31	5.71	16.27	91.13	31.77	70.56	2.86
	Skewness	0.62	-0.59	0.92	3.22	8.71	4.68	7.47	-0.58
Hybrid	**Mean**	94.91	32.34	0.38	0.53	0.13	588.00	0.09	0.77
	MAD	15.36	5.11	0.24	0.30	0.14	779.20	0.11	0.16
	Median	95.73	35.14	0.31	0.57	0.05	42.00	0.03	0.79
	Mode	64.05	18.98	0	0.04	0	0	0	0.79
	SD	19.56	6.52	0.29	0.37	0.18	913.94	0.15	0.22
	Kurtosis	1.93	2.85	1.98	1.59	2.56	2.08	3.12	3.05
	Skewness	-0.31	-0.87	0.27	-0.06	1.02	0.98	1.39	-0.86
Sybil	**Mean**	94.43	25.16	0.36	0.50	0.13	2228.19	0.10	0.65
	MAD	20.52	8.60	0.34	0.44	0.17	3151.83	0.14	0.36
	Median	92.04	26.89	0.20	0.45	0.02	100	0	0.89
	Mode	36.67	0	0	0	0	0	0	1
	SD	24.79	10.49	0.49	0.52	0.27	5247.83	0.21	0.39
	Kurtosis	2.24	2.68	8.53	2.90	7.82	16.12	7.52	1.60
	Skewness	0.03	-0.64	2.21	0.79	2.47	3.47	2.38	-0.58

Appendix B. Classifier configurations

Table B.10 shows the parameter settings for the four machine learning algorithms employed in this study.

Table B.10
Parameters used in different classifiers.

Parameter	Classification models								
	Decision tree				Bayes network	SVM		Multilayer artificial neural network	
Training Tool	Weka 3.7.6								
Algorithm	J48 (C4.5)	Logistic Model Tree	Random Forest	LogitBoost	Bayesian Network	SMO (SVM)		Multilayer perceptron trained with the gradient descent method (MLP-GD)	
Abbreviation	J48	LMT	RFT	LBT	BayesNet	SMO-P	SMO-R	MLP-GD10	MLP-GD20
Classifier				Decision Stump	Simple estimator				
Search Algorithm					Simulated annealing			Gradient descent	
Confidence Factor	0.25								
Number of trees			10						
Maximum depth			unlimited						
Minimum number of instances per leaf	2								
Pruning	Yes								
Validation technique	Cross-validation with 10 folds								
Training time								500 epochs	
Learning rate								0.3	
Momentum								0.2	
Data normalization						Yes		Yes	
Number of hidden layers								1	
Neurons in the hidden layer								10	20
Activation function								sigmoid	
Validation threshold								20	
Complexity parameter (c)						1			
Kernel function						Poly.	RBF		
others	kept as default								

References

[1] A. Alarifi, M. Alsaleh, Web spam: A study of the page language effect on the spam detection features, in: Machine Learning and Applications (ICMLA), 2012 11th International Conference on, vol. 2, IEEE, 2012, pp. 216–221.
[2] A. Alarifi, M. Alsaleh, A. Al-Salman, A. Alswayed, A. Alkhaledi, Google penguin: Evasion in non-english languages and a new classifier, in: Machine Learning and Applications (ICMLA), 2013 12th International Conference on, vol. 2, IEEE, 2013, pp. 274–280.
[3] A. Almaatouq, A. Alabdulkareem, M. Nouh, E. Shmueli, M. Alsaleh, V.K. Singh, A. Alarifi, A. Alfaris, A.S. Pentland, Twitter: who gets caught? observed trends in social micro-blogging spam, in: Proceedings of the 2014 ACM Conference on Web Science, ACM, 2014, pp. 33–41.
[4] M. Alsaleh, A. Alarifi, A.S. Al-Salman, M. Alfayez, A. Almuhaysin, TSD: detecting sybil accounts in twitter, in: 13th International Conference on Machine Learning and Applications, ICMLA 2014, Detroit, MI, USA, December 3–6, 2014, pp. 463–469.
[5] M. Alsaleh, P. van Oorschot, Evaluation in the absence of absolute ground truth: toward reliable evaluation methodology for scan detectors, Int. J. Inf. Secur. 12 (2) (2013) 97–110.
[6] L. Alvisi, A. Clement, A. Epasto, S. Lattanzi, A. Panconesi, Sok: The evolution of sybil defense via social networks, in: Security and Privacy (SP), 2013 IEEE Symposium on, IEEE, 2013, pp. 382–396.
[7] I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, G. Paliouras, C.D. Spyropoulos, An evaluation of naive bayesian anti-spam filtering, arXiv preprint cs/0006013 (2000).
[8] L. Becchetti, C. Castillo, D. Donato, S. Leonardi, R.A. Baeza-Yates, Link-based characterization and detection of web spam., in: AIRWeb, 2006, pp. 1–8.
[9] F. Benevenuto, G. Magno, T. Rodrigues, V. Almeida, Detecting spammers on twitter, in: Collaboration, electronic messaging, anti-abuse and spam conference (CEAS), 6, 2010, p. 12.
[10] A. Beutel, W. Xu, V. Guruswami, C. Palow, C. Faloutsos, Copycatch: Stopping group attacks by spotting lockstep behavior in social networks, in: Proceedings of the 22Nd International Conference on World Wide Web, in: WWW '13, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 2013, pp. 119–130.
[11] Y. Boshmaf, I. Muslukhov, K. Beznosov, M. Ripeanu, The socialbot network: when bots socialize for fame and money, in: Proceedings of the 27th Annual Computer Security Applications Conference, ACM, 2011, pp. 93–102.
[12] Q. Cao, M. Sirivianos, X. Yang, T. Pregueiro, Aiding the detection of fake accounts in large scale social online services, in: Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12), 2012, pp. 197–210.
[13] Q. Cao, X. Yang, J. Yu, C. Palow, Uncovering large groups of active malicious accounts in online social networks, in: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2014, pp. 477–488.
[14] Z. Chu, S. Gianvecchio, H. Wang, S. Jajodia, Detecting automation of twitter accounts: Are you a human, bot, or cyborg? Dependable Secure Comput. IEEE Trans. 9 (6) (2012) 811–824.
[15] H. Drucker, S. Wu, V.N. Vapnik, Support vector machines for spam categorization, Neural Netw. IEEE Trans. 10 (5) (1999) 1048–1054.
[16] M. Erdélyi, A. Garzó, A.A. Benczúr, Web spam classification: a few features worth more, in: Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality, 2011, pp. 27–34.
[17] D. Fetterly, M. Manasse, M. Najork, Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages, in: Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004, ACM, 2004, pp. 1–6.
[18] D. Fetterly, M. Manasse, M. Najork, Detecting phrase-level duplication on the world wide web, in: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2005, pp. 170–177.
[19] C. Freitas, F. Benevenuto, S. Ghosh, A. Veloso, Reverse engineering socialbot infiltration strategies in twitter, in: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, ACM, 2015, pp. 25–32.
[20] S.A. Golder, M.W. Macy, Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures, Science 333 (6051) (2011) 1878–1881.
[21] Z. Gyöngyi, H. Garcia-Molina, J. Pedersen, Combating web spam with trustrank, in: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, VLDB Endowment, 2004, pp. 576–587.
[22] Z. Gyöngyi, G.-M. H., Web spam taxonomy, in: Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web, 2005.
[23] M.A. Hall, Correlation-based Feature Subset Selection for Machine Learning, University of Waikato, Hamilton, New Zealand, 1998 Ph.D. thesis.
[24] R. Holt, Twitter in numbers, 2013, Accessed: July 2014. http://www.telegraph.co.uk/technology/twitter/9945505/Twitter-in-numbers.html.
[25] T. Hwang, I. Pearce, M. Nanis, Socialbots: Voices from the fronts, interactions 19 (2) (2012) 38–45.
[26] I. Jolliffe, Principal Component Analysis, Springer Series in Statistics, Springer, 2002 URL http://books.google.com.sa/books?id=_olByCrhjwIC..
[27] H.-S. Kim, W.-Y. Shin, D. Kim, J. Cho, Improved tweet bot detection using spatio-temporal information, J. Korea Inst. Inf. Commun. Eng. 19 (12) (2015) 2885–2891.
[28] S.B. Kim, P. Rattakorn, Unsupervised feature selection using weighted principal components, Expert Syst. Appl. 38 (5) (2011) 5704–5710.
[29] J.R. Landis, G.G. Koch, The measurement of observer agreement for categorical data, biometrics (1977) 159–174.
[30] A.-C. Lee, G.-E. Seo, W.-Y. Shin, D. Kim, J. Cho, Improved tweet bot detection using geo-location and device information, J. Korea Inst. Inf. Commun. Eng. 19 (12) (2015) 2878–2884.
[31] K. Lee, B.D. Eoff, J. Caverlee, Seven months with the devils: A long-term study of content polluters on twitter., ICWSM, Citeseer, 2011.
[32] W. Ltd, Twitter4j - a java library for the twitter api, 2013, Accessed: July 2014. http://twitter4j.org/en/index.html.
[33] J. Messias, L. Schmidt, R. Oliveira, F. Benevenuto, You followed my bot! transforming robots into influential users in twitter, First Monday 18 (7) (2013).
[34] C.C. Miller, The loyal users of google plus say it is no ghost town, 2014, Accessed: July 2014. http://bits.blogs.nytimes.com/2014/02/19/the-loyal-users-of-google-plus-say-it-is-no-ghost-town/?_php=true&_type=blogs&_r=0.
[35] A. Ntoulas, M. Najork, M. Manasse, D. Fetterly, Detecting spam web pages through content analysis, in: Proceedings of the 15th international conference on World Wide Web, ACM, 2006, pp. 83–92.
[36] A. Paradise, R. Puzis, A. Shabtai, Anti-reconnaissance tools: Detecting targeted socialbots, IEEE Internet Comput. 18 (5) (2014) 11–19.
[37] J. Piskorski, M. Sydow, D. Weiss, Exploring linguistic features for web spam detection: A preliminary study, in: Proceedings of the 4th international workshop on Adversarial information retrieval on the web, ACM, 2008, pp. 25–28.
[38] N.M. Radziwill, M.C. Benton, Bot or not? deciphering time maps for tweet interarrivals, arXiv preprint arXiv:1605.06555(2016).
[39] J. Ratkiewicz, M. Conover, M. Meiss, B. Gonçalves, S. Patil, A. Flammini, F. Menczer, Truthy: Mapping the spread of astroturf in microblog streams, in: Proceedings of the 20th International Conference Companion on World Wide Web, in: WWW '11, ACM, New York, NY, USA, 2011, pp. 249–252.
[40] U.S. Securities, E. Commission, Amendment no. 1 to form s-1, 2013, Accessed: July 2014. http://www.sec.gov/Archives/edgar/data/1418091/000119312513400028/d564001ds1a.htm#toc564001_11.
[41] K.M. Svore, Q. Wu, C.J. Burges, A. Raman, Improving web spam classification using rank-time features, in: Proceedings of the 3rd international workshop on Adversarial information retrieval on the web, ACM, 2007, pp. 9–16.
[42] D.N. Tran, B. Min, J. Li, L. Subramanian, Sybil-resilient online content voting., in: NSDI, vol. 9, 2009, pp. 15–28.
[43] N. Tran, J. Li, L. Subramanian, S.S. Chow, Optimal sybil-resilient node admission control, in: INFOCOM, 2011 Proceedings IEEE, IEEE, 2011, pp. 3218–3226.
[44] I. Twitter, Rest api v1.1 limits per window by resource, 2013, Accessed: July 2014. https://dev.twitter.com/docs/rate-limiting/1.1/limits.
[45] B. Viswanath, A. Post, K.P. Gummadi, A. Mislove, An analysis of social network-based sybil defenses, ACM SIGCOMM Comput. Commun. Rev. 40 (4) (2010) 363–374.
[46] C. Wagner, S. Mitter, C. Körner, M. Strohmaier, When social bots attack: Modeling susceptibility of users in online social networks, Making Sense Microposts (# MSM2012) 2 (2012).

[47] A.H. Wang, Don't follow me: Spam detection in twitter, in: Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on, 2010, pp. 1–10.
[48] G. Wang, T. Konolige, C. Wilson, X. Wang, H. Zheng, B.Y. Zhao, You are how you click: Clickstream analysis for sybil detection, in: Proceedings of the 22Nd USENIX Conference on Security, in: SEC'13, USENIX Association, Berkeley, CA, USA, 2013, pp. 241–256.
[49] G. Wang, M. Mohanlal, C. Wilson, X. Wang, M. Metzger, H. Zheng, B.Y. Zhao, Social turing tests: Crowdsourcing sybil detection, arXiv preprint arXiv:1205.3856(2012).
[50] Y. Xie, F. Yu, Q. Ke, M. Abadi, E. Gillum, K. Vitaldevaria, J. Walter, J. Huang, Z.M. Mao, Innocent by association: early recognition of legitimate users, in: Proceedings of the 2012 ACM conference on Computer and communications security, ACM, 2012, pp. 353–364.
[51] Z. Yang, C. Wilson, X. Wang, T. Gao, B.Y. Zhao, Y. Dai, Uncovering social network sybils in the wild, ACM Trans. Knowl. Discovery from Data (TKDD) 8 (1) (2014) 2.
[52] H. Yu, P.B. Gibbons, M. Kaminsky, F. Xiao, Sybillimit: A near-optimal social network defense against sybil attacks, in: 2008 IEEE Symposium on Security and Privacy (sp 2008), IEEE, 2008, pp. 3–17.
[53] E. Zangerle, G. Specht, Sorry, i was hacked: A classification of compromised twitter accounts, in: Proceedings of the 29th Annual ACM Symposium on Applied Computing, ACM, 2014, pp. 587–593.