Features combination for the detection of malicious Twitter accounts

Isaac David, Oscar S. Siordia, Daniela Moctezuma
Centro de investigación en Geografía y Geomática Ing. Jorge L. Tamayo AC
México D.F., Mexico
Email: isacdaavid@isacdaavid.info, {osanchez,dmoctezuma}@centrogeo.edu.mx

***Abstract*—Microblogging social networks are easily subverted by automated fake identities that amass disproportionately large influence. In this paper we present an effort to profile and screen such kind of accounts from existing and original ground truth obtained from the Twitter platform. Seventy-one explanatory properties solely extracted from profile and timeline information are evaluated and used to compare the efficacy of common supervised machine learning methods at this classification task. Results confirm that feasible and largely effective detection devices can be constructed for the problem at hand.**

I. Introduction

Microblogging is a form of Web publishing characterized by very small text entries in the order of a few sentences, navigated in reverse chronological order and sometimes enhanced with rich media. Since the late 2000's, distributed systems have been created to provide services and networks for users who want to author a microblog or interact with other microbloggers.

Together with its popularity; the ability to influence featured taglines, promote content on user timelines and inflate profiles has made of Twitter an attractive microblogging platform to marketers and deceivers, who commonly employ fake accounts to achieve their goals. For greater impact, campaign managers and popularity builders orchestrate legions of hundreds or thousands of Sybil accounts that do nothing but repeat propaganda [1]. Automated accounts known as "Twitterbots" or simply "bots" are a tangential phenomenon to content pollution and Sybil armies. Some strands of bots produce useful content or serve as unoriginal and repetitive, yet non-malicious content aggregators and personal assistants (sometimes even coexisting with user-generated posts in a what is known as a hybrid or "cyborg" account). The use of automated accounts, however, greatly amplifies the influence of spammers, and contributes to the amount of rather uninteresting and undesirable information in Internet communities.

News about celebrities and political parties buying influence from bot army operators abound, in tune with the findings of external researchers and the testimony of organizations like Twitter and Renren [1] [2]. We argue the consequences of bots in social networks are two-fold, at best: presuming an economic incentive behind operating Sybil armies, insofar as those are tolerated by real users, the social network thrives and the incentive for bots is kept; but even in balance some parties will result invariably affected. For one, some platforms sell their own ad space and therefore compete in their own game with bot armies that are available for rental. Investors and legitimate advertisers are wary of targeting their efforts at a mixed bag of real and fake accounts. And of course, users and external observers are presented with a false impression of the size and dynamics of a social network; something that could easily pass unnoticed to the untrained eye. When the influence of an Internet social network is so profound that a considerable portion of the population gets its news from there, and even traditional journalism starts referencing back to it; bot-based disruption operations of the scope and scale witnessed to this day become a danger that should not be dealt with lightly. The motivation behind this study stems from our own necessity to filter out unreliable Twitter data, collected for the purpose of performing other studies.

In this paper we explore the construction and evaluation of an automated Turing test to tell computers and humans apart (the "ATCHA" in the acronym CAPTCHA [3]), albeit one designed for a narrow form of artificial agents common to Twitter and similar microblogging social networks. We employ and compare a number of supervised machine learning techniques for this purpose. An intelligent agent is said to be learning if it improves its performance on future tasks after making observations about the world [4]. Learning algorithms allow computer programs capable of prediction and pattern recognition to be built empirically; that is, from illustrative data and no *a priori* knowledge like explicit rules.

The paper is organized as follows: section II talks about previous research and the state of the art in bot detection. Section III describes our sample data and their origins. Section IV elaborates on feature extraction and model selection. Finally, section V evaluates the detection system and provides conclusions.

II. Related Work

Some work has been done to estimate the amount of fraudulent user accounts on Twitter. In its 2013, 2014 and 2015 annual reports to the United States Securities and Exchange Commission; Twitter, Inc. claimed that Spam and fake accounts represented about 5% of all monthly active users (based on a sample of internally reviewed accounts). Quoting from there, Spam is defined as *"unsolicited, repeated actions that negatively impact other users with the general goal of drawing attention"*. Monthly active users who accessed the

978-1-5090-3794-0/16/$31.00

site through third-party applications (as opposed to the website or official mobile clients) were estimated to be 7%, 8.5% and 8.5% in 2013, 2014 and 2015; respectively [5] [6] [7]. Third-party applications are possible thanks to Twitter's public application programming interface (API for short).

These two measures —Spam and API-using accounts— are important in the study of bot detection because evidence suggests that most Spam on Twitter is the product of Sybils relying on third-party tools to communicate to Twitter via the API [8]. Therefore, Spam and API-using accounts could respectively serve as rough surrogates for the lower and upper bounds to the bot-to-human ratio on Twitter.

The connection between Spam and bots has been exploited in the past by similar supervised machine learning approaches at bot detection. In 2010, Wang et al. studied the application of Bayesian classifiers commonly found in e-mail Spam detection to recognize unsolicited messages in the early Twitter [9] [10]. In 2012, Chu et al. carried out an extensive study on the characterization of Twitter bots and cyborgs, and resorted to supervised machine learning methods to build a ternary classifier with an average accuracy of 96% upon cross-validation. 10.5% of their full dataset (consisting of over half a million accounts) was predicted to be a bot according to this classifier [8].

Using a labeled dataset by Lee et al. from 2011 which contains some tens of thousands of potential spammers, legitimate users and their tweets [11]; Ferrara et al. trained a binary classifier that could predict with 95% accuracy the class those accounts belonged to. [12] [13]. This system is available online as a web-service. Lee and colleagues' dataset also proved useful as a starting point to some of the contestants in the Twitterbot recognition competition organized by the US Defense Advanced Research Projects Agency (DARPA) in 2015 [14].

Similarly, Alsaleh et al. and Yang et al. obtained good results for Twitter in the former's case, and for an equivalent Chinese microblogging network called Sina in the latter's [15] [16]. Cerón-Guzmán et al. presented a similar effort for detecting spammers during the 2014 Colombian presidential election, also with promising results [17].

Most machine learning studies mentioned so far revolve around the use of supervised classifiers on features pertaining to account information and posting activity patterns, and less prominently to interaction graphs and simple linguistic cues or blacklisted elements buried within posts. However, parallel lines of research build more prominently on statistical analysis of language [18] and sentiment analysis [19] in conjunction with supervised methods; while others like [20] have explored the possibility of performing semi-automatic clustering and ground-truth creation, based on the same user account and tweeting features.

III. Data Collection and Ground Truth Creation

We leveraged some of the labeling work done by anonymous project BotsDeTwitter[1] (@BotsPoliticosNo) to build half of our dataset. BotsDeTwitter is a blog with a relentless record of uncovering and documenting Sybil networks from the Spanish-speaking world, mostly in the context of Spain politics. Rather than sampling at random, the method used by BotsDeTwitter begins by taking notice of election front-runners and their social surroundings, for instance their retweeters. These users' accounts and part of their timelines are extracted via the Twitter API. Posts are plotted in time to reveal account creation patterns and uncannily synchronized retweeting patterns (see Fig 1). Social graphs reveal anomalies that trace back not only to whomever is requesting or benefiting from the service, but sometimes also to army operators themselves who don't hesitate to advertise their powers and inflate their own personas. Moreover, account samples are individually verified, just to find out that everything; from profile pictures taken from Web searches to the strange and static choice of automatable application programs used to post, conforms to the fakery hypothesis.

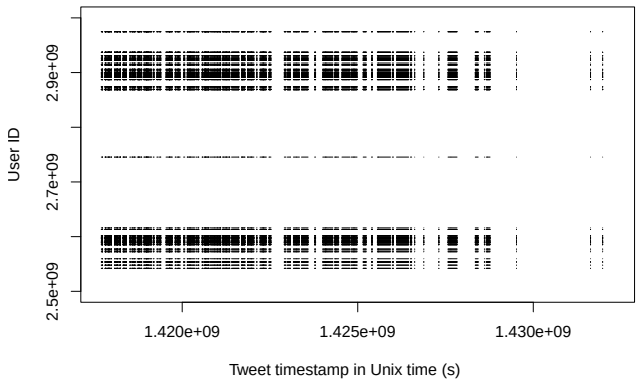

Fig. 1. Six months worth of data from 142 Sybil timelines in our dataset, working in synchronicity to leave a common activity trail in time.

We started with 1598 different Sybil accounts belonging to five different armies from Spain and Argentina that were detected throughout the second half of 2015. Apart from naming and shaming, BotsDeTwitter's ultimate goal is to report bot accounts to be taken down, further diminishing the chances of finding them online. For this reason we had to settle with a final sample of 853 bot profiles and the latest 1000 tweets in each one's timeline. They were extracted over the course of one week. This was complemented with 791 human accounts manually labeled between April and June 2016, most of which belong to Mexican users.

IV. Feature Engineering

User profiles, their timelines and network interactions offer three different information sources for feature generation. In our study only user profiles and timelines were used to generate an initial feature set of 71 inexpensive variables. We further separate timeline-based features into metadata-based and content-based ones. Metadata refers to all acces-

[1]https://botsdetwitter.wordpress.com

sory information supporting or describing the main content. Such a number of features may appear excessive on first account, but many of them naturally emerge from the calculation of standard statistical estimators (such as centrality and dispersion measures) of a smaller number of random distributions hidden in the data. These distributions include tweet interrarival times, tweets ranging over different time periods, over posting sources; and the number of interactive elements (URLs, hashtags and mentions) per tweet. In fact our feature set is far from complete, specially in the area of content-based measures, where statistical analysis of language alone enjoys much wider use. Table I presents in more detail all candidate explanatory variables considered for this study.

Of interest is Shannon entropy —entropy henceforth— which measures a discrete distribution's theoretical pure information content; and is given by the expected value of the "surprisal" caused by the outcomes of the random variable at play:

$$H(X) = E[-\log_b(P(X))] = -\sum_{i=1}^{n} P(x_i)\log_b(P(x_i)). \quad (1)$$

where $E[Y]$ is the expected value function and $P(x_i)$ the probability mass function of discrete random variable X. The choice of b only determines what units go into the equation. $H(X)$ is in bits for $b = 2$ [21].

Besides estimating the amount of information contained in strings, entropy proves useful to distinguish humans from bots based on their temporal activity patterns. The underlying idea is that bot activity will be more predictable (or perhaps completely random if programmed to do so), whereas humans will exhibit less strict behaviors.

A word of caution: variables which are more amenable to continuous distributions such as interarrival times (whose accuracy is measured to the second) as well as anemic samples from truly discrete distributions (e.g. a few tweet timestamps distributed over the days of the year) are immune to the statistical formulation of entropy, which by definition depends on a representative sample in order to be a good estimator of information content. For this reason domain-specific lossless compression algorithms are sometimes brought in, *in lieu* of entropy, to address the same question of information content.

A. Exploring the data

Unlike many other classification models, decision trees have a representation that is transparent to people; they are easy to interpret and offer an attractive tool to grow new insights on how the two user account classes differ in high-dimensional feature space. Although preliminarily, we can offer a brief recollection of some important differences that were discovered while feeding computed features to decision trees.

In accordance to [8], account creation time alone constitutes a big difference between bots and humans in the data; not just because microblogging bots are a more recent invention that

TABLE I
LIST OF FEATURES

	Feature	Details
user profile	created at [8], [12], [13]	Account creation time
	default profile image [14], [15]	Whether account uses a stock image.
	description entropy	See (1)
	description hashtags count	Amount of hashtags in description
	description length	[2]
	description mentions count	Amount of mentions in description
	description URLs count	Amount of URLs in description
	empty description [17]	Whether description is empty
	empty location	Whether location is empty
	empty URL	Whether URL is empty
	statuses count [8], [12]–[14], [16]	[2]
	favourites count [15]	[2]
	favourites per status	[2]
	favourites per time	$favourites/account\_age$
	followers count [17]	[2]
	followers per friend [8], [15]	[2]
	followers per time	$followers\_count/account\_age$
	friends count [17]	[2]
	friends per time [17]	$friends\_count/account\_age$
	lang [12], [13]	User locale
	listed count	Number of lists the user is part of
	listed count per followers	$listed\_count/followers\_count$
	listed count per time	$listed\_count/account\_age$
	location entropy	See (1)
	location length	[2]
	name entropy	See (1)
	name length [12], [13]	[2]
	same-description user proportion	Ratio of users sharing same description
	same-location user proportion	Ratio of users sharing same location
	same-name user proportion	Ratio of users sharing same name
	same-screen name user proportion	Ratio of users sharing same screen name
	same-URL user proportion [14]	Ratio of users sharing same URL
	screen name entropy	See (1)
	screen name length [12], [13]	[2]
	statuses per time	$statuses\_count/account\_age$
	verified [8], [17]	Is the verification badge present?
timeline metadata	automated sources proportion	See [8], [14], [17]
	delay mean [8], [12], [13], [17]	Average interrarival time
	delay S.D. [8], [12], [13], [17]	Interrarival time (std. dev.)
	ms delay mean [12], [13]	Original status delay (average)
	ms delay S.D. [12], [13]	Original status delay (std. dev.)
	ms proportion [12]–[15], [17]	Original statuses proportion
	number of sources [8], [14]	Number of different posting sources
	tweets-mod-day S.D [8], [17]	Hour of the day (std. dev.)
	tweets-mod-week entropy [8], [17]	Day of the week (entropy)
	tweets-mod-week mean [8], [17]	Day of the week (average)
	tweets-mod-week S.D. [8], [17]	Day of the week (std. dev.)
	tweets-mod-year entropy	Day of the year (entropy)
	tweets-mod-year mean	Day of the year (average)
	tweets-mod-year S.D.	Day of the year (std. dev.)
	tweets-per-source mean [8], [14]	Average number of statuses per source
	tweets-per-source S.D. [8]	Statuses per source (std. dev.)
	tweets-per-source entropy	Statuses per source (entropy)
	replies proportion [12]–[15], [17]	Replies proportion
	reply delay mean [12], [13]	Reply delay (average)
	reply delay S.D. [12], [13]	Reply delay (std. dev.)
	retweet delay mean [12], [13]	Repost interarrival time (average)
	retweet delay S.D. [12], [13]	Repost interarrival time (std. dev.)
	retweets proportion [12]–[15], [17]	Repost proportion
timeline content	hashtags-per-tweet entropy	See (1)
	hashtags-per-tweet mean	See [14], [15], [17], [20]
	hashtags-per-tweet S.D.	[2]
	mentions-per-tweet entropy	See (1)
	mentions-per-tweet mean	See [14], [15], [17], [20]
	mentions-per-tweet S.D.	[2]
	tweets-with-hashtags proportion	See [8]
	tweets-with-mentions proportion	See [8]
	tweets-with-URLs proportion	See [8]
	URLs-per-tweet entropy	See (1)
	URLs-per-tweet mean	See [14], [15], [17], [20]
	URLs-per-tweet S.D.	[2]

[2]The meaning of these features should be obvious from their names.

only appeared after the popularization of microblogging sites. Malicious accounts need to be replenished more frequently to cope with the janitorial endeavours going on in the platform.

Timelines where less than 1.8% of all tweets were replies were 5.56 times more likely to belong to a Sybil. Moreover, in 64% of all Sybil profiles the proportion of replies was less than 0.5%; suggesting that these accounts do not engage in conversation in any substantive way. An account with less than 1.8% of replies and less than 432 followers was almost guaranteed to belong to a Sybil. On the other hand, some famous human and organizational accounts don't engage in much conversation either.

Interestingly, Sybils on average presented more spread out and entropic distributions of interactive elements per tweet. For instance, a bot's distribution of mentions per tweet is 7.6 times as likely as a human's to offer more than 2.3 bits. This could mean that humans almost never try to go past two or three mentions per tweet, even if on average they might insert as many mentions as bots do. High proportion of tweets with hashtags is indicative of Sybil behavior too.

B. Variable Importance

Effectively finding the best feature combination from a feature set $features$ (by exhausting its power set) takes $\Omega(2^{|features|})$ steps. Consequently, combinatorial explosion readily renders the problem intractable for cardinalities as big as ours. Machine learning as a field is constantly pushing to find novel ways of doing feature learning, and artificial intelligence has developed a number of search methods to help approximate solutions to optimization problems. Our approach to reduce dimensionality consists in ordering features by some sort of predictive relevancy before they are put to test in a model. Thus all features were ranked by importance according to four distinct measures:

- The RELIEF feature selection algorithm for classification problems (for a sample size of 40 instances and 20 neighbors per instance) [22].
- Variable importance according to a single decision tree trained with full data.
- Mean accuracy decrease and
- mean Gini importance from the random forest probabilistic algorithm, [23] [24] also trained to classify the whole dataset.

A random forest is an ensemble of different decision trees, each one casting its vote to reach a consensus at the forest level. High mean accuracy decrease and high mean Gini decrease translate into variable importance. In short, determining decrease in accuracy involves eliminating or permuting a feature to observe changes to the classification error. As for decrease in Gini node impurity, whenever a node is split, its Gini importance is calculated from the children nodes and compared against their own coefficients. A random forest implementation can compute both mean accuracy and Gini decreases transparently during the construction phase of the classifier.

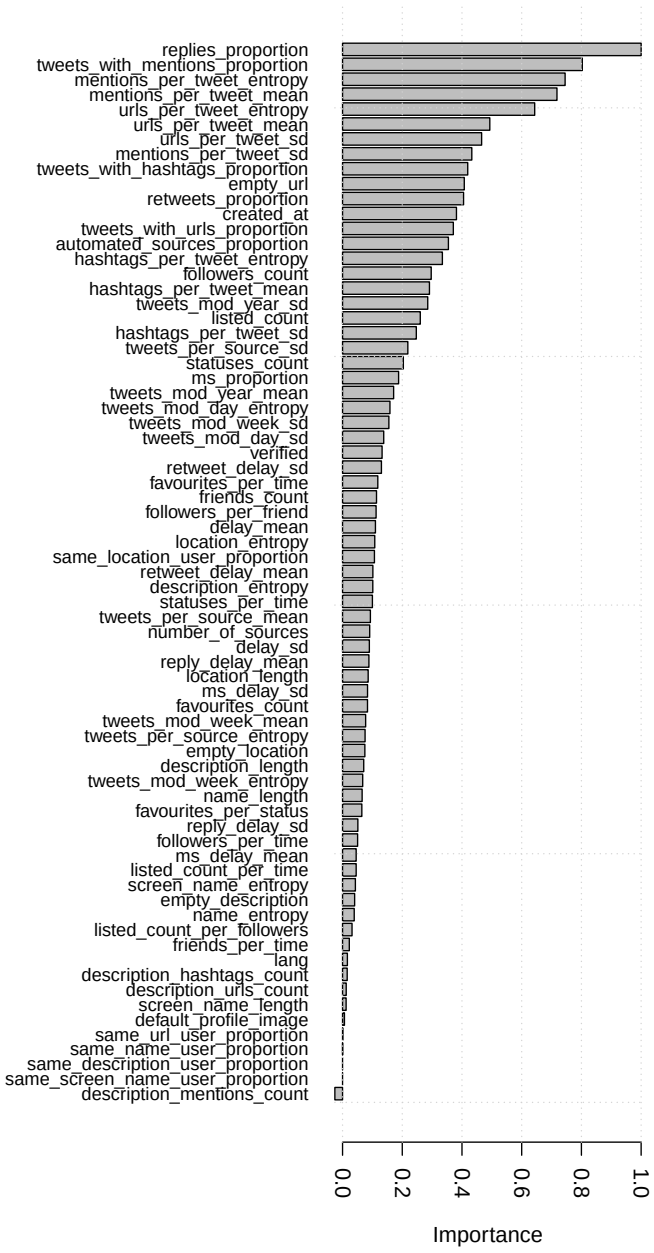

Fig. 2. Normalized average variable importance.

Finally, we took the average variable importance from the four previous rankings, under the assumption that features which are favorably selected for by more methods are more likely to provide explanatory power to a variety of these. Scores were normalized to a common scale prior to calculating the arithmetic mean so that no contribution is favored. Results are shown in Fig. 2.

C. Selection by Validated Classification

Once features had been sorted by relevance, we proceeded to assess the accuracy of five types of supervised classifiers: support vector machines, decision trees, naive Bayes,

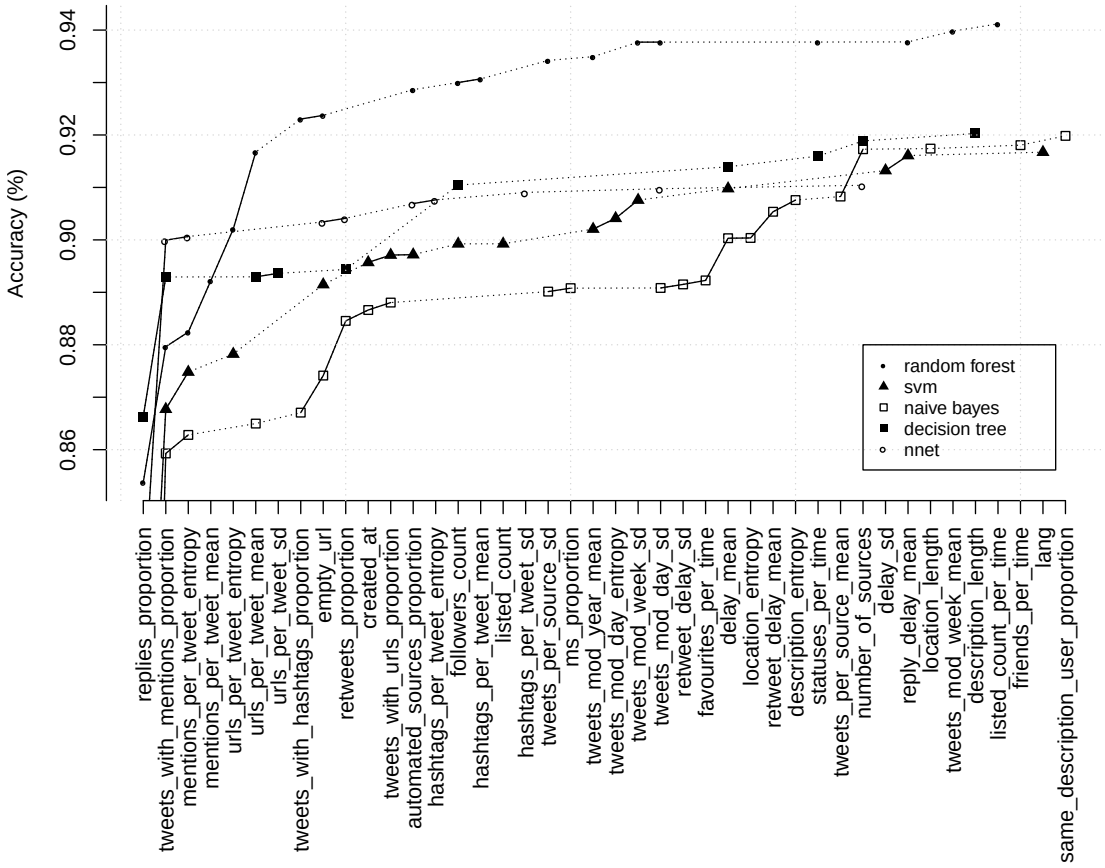

Fig. 3. Accuracy as more features of decreasing importance are considered.

random forest and single-hidden-layer feedforward artificial neural networks (also known as the multilayer perceptron). [4] [25] Linear and radial kernels were tested for support vector machines, but only linear ones will be considered below since they obtained better results. Decision trees were obtained using a recursive partitioning algorithm and pruned to avoid overfitting.

On the first run all classifiers were trained using the single best variable; accuracy was measured using 10-fold cross-validation. In each following round the next best variable was added to the mix, and was preserved only by those classifier methods whose accuracy did not shrink from the appearance of the new variable (also using a 10-fold cross-validation to measure accuracy and compare results). Otherwise the offending variable was never reconsidered for that particular classifier. This sort of greedy hill climbing guided by a feature ordering not only guarantees stable improvement on previous results, but also seemed to improve some of the classifiers' long-term performance with no visible downsides. Naive Bayes' best result used to be 3% lower when allowed to accumulate destructively-interfering variables, and accuracy quickly plummeted after a few ones. Since variables that are more likely strongly correlated with the dependent class are considered first, we infer there's little room for improvement between our results and a hypothetical, globally optimum variable selection.

Results are shown in Fig. 3. Features are ranked by decreasing importance from left to right: at any given point all classifiers have been exposed to all features to the left of that point, inclusive. Note that this allows meaningful benchmarking among classifiers, with one caveat: as mentioned before, the exact feature combination leading to a result doesn't include features that were deemed counterproductive when first considered. These missing features are shown as dotted line segments in the plots. Features that negatively impacted all five methods were omitted for brevity.

V. Results

From Fig. 3 we observe that the highest average accuracy (94%) was obtained with a random forest operating on 19 features, although gains were not as dramatic after the first 6. A breakdown of both error types for this particular model appears in the confusion matrix at table II.

The relatively low variability of results across classifiers reinforces the idea that feature quality and feature subset selection are playing an important role in accurate prediction. Despite converging towards 91%-92%, the remaining methods

TABLE II
CONFUSION MATRIX (RANDOM FOREST)

Prediction	Human	Bot
Human	677	64
Bot	19	669

did not do similarly well in terms of growth; for naive Bayes required 23 features to reach an accuracy level similar to a decision tree which could get there using only 10 of them. The short-lived supremacy and subsequent stagnation of neural networks was probably due to an increase in input neurons, which pushed training phases to the predefined time limit. Even so, neural networks had the most consuming learning process of all classifiers.

Later on, the aforementioned winner was used to provide a response to 5063 fresh and unlabeled Mexican accounts. 13.5% were found to be potential bots. Upon manual inspection of a loose sample of results we found ourselves seldom diverting from the automatic decision. Heavy retweeters remain typical among suspect Sybils, and many of them unsurprisingly relayed great amounts of politically opinionated, partisan messages from prominent political stakeholders during the last Mexican elections cycle. Accounts set up to automatically forward repetitive notifications —from activity reported by places like Youtube or Instagram— also form a large subgroup within the identified accounts.

At the time of writing we are tweaking the last details of a Web service, available at datanlab.com, and backed by this classifier. We hope to provide a solution to any individual or organization interested in detecting Twitter Sybil accounts, starting from our geographic, cultural and political neighborhood.

REFERENCES

[1] Z. Yang, C. Wilson, T. Gao, B. Y. Zhao, and Y. Dai, "Uncovering social network sybils in the wild," 2011. [Online]. Available: https://arxiv.org/pdf/1106.5321.pdf

[2] E. Treré, "The dark side of digital politics: Understanding the algorithmic manufacturing of consent and the hindering of online dissidence," *IDS Bulletin*, vol. 47, no. 1, 2016. [Online]. Available: http://bulletin.ids.ac.uk/idsbo/article/view/41

[3] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, *CAPTCHA: Using Hard AI Problems for Security*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 294–311. [Online]. Available: http://dx.doi.org/10.1007/3-540-39200-9_18

[4] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Pearson Education, 2009.

[5] Twitter, Inc., "Annual report 2013," Twitter, Inc., 1355 Market Street, Suite 900, San Francisco CA, Tech. Rep., 2013. [Online]. Available: http://files.shareholder.com/downloads/AMDA-2F526X/2419071766x0x742484/A418947A-E065-4822-8BD4-00FA8EB4E795/Twitter_2013_Annual_Report_-_FINAL.pdf

[6] ——, "Annual report 2014," Twitter, Inc., 1355 Market Street, Suite 900, San Francisco CA, Tech. Rep., 2014. [Online]. Available: http://files.shareholder.com/downloads/AMDA-2F526X/2419071766x0x821792/AB7B5D27-2CEB-468B-9C9B-469676E3186A/876564_Twitter_Annual_Report.pdf

[7] ——, "Annual report 2015," Twitter, Inc., 1355 Market Street, Suite 900, San Francisco CA, Tech. Rep., 2016. [Online]. Available: http://files.shareholder.com/downloads/AMDA-2F526X/2419071766x0x886152/3FBBB0EC-FDF0-41D2-9C4E-A06AE8B1D1E5/2016_Twitter_Annual_Report.pdf

[8] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Detecting automation of Twitter accounts: Are you a human, bot, or cyborg?" *IEEE Trans. Dependable Secure Comput.*, vol. 9, 2012.

[9] A. H. Wang, "Don't follow me: Spam detection in Twitter," in *IEEE International Conference on Security and Cryptography*, 2010, pp. 1–10.

[10] ——, "Detecting spam bots in online social networking sites: A machine learning approach," in *Data and Applications Security and Privacy XXIV*, 2010, pp. 335–342. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-13739-6_25

[11] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on Twitter," in *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*, Barcelona, Spain, 2011.

[12] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, "The rise of social bots," 2015. [Online]. Available: http://arxiv.org/pdf/1407.5225v3.pdf

[13] ——, "BotOrNot: A system to evaluate social bots," 2016. [Online]. Available: http://arxiv.org/pdf/1602.00975v1.pdf

[14] V. Subrahmanian, A. Azaria, S. Durst, K. Vadim, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer, "The DARPA Twitter bot challenge," 2016.

[15] M. Alsaleh, A. Alarifi, A. M. Al-Salman, and M. Alfayez, "TSD: Detecting sybil accounts in Twitter," in *13th International Conference on Machine Learning and Applications*, 2014.

[16] W. Yang, G. Dong, W. Wang, H. YX., G. Shen, and M. Yu, "A novel approach for bots detection in Sina microblog," *Journal of Computational and Theoretical Nanoscience*, vol. 12, pp. 1420–1425, 2015.

[17] J. A. Cerón-Guzmán and E. León, "Detecting social spammers in Colombia 2014 presidential election," in *Advances in Artificial Intelligence and Its Applications: 14th Mexican International Conference on Artificial Intelligence, MICAI 2015, October 25-31, 2015, Proceedings, Part II*, O. Pichardo Lagunas, O. Herrera Alcántara, and G. Arroyo Figueroa, Eds. Cuernavaca, Mexico: Springer International Publishing, 2015, pp. 121–141.

[18] J. Martínez-Romo and L. Araujo, "Detecting malicious tweets in trending topics using a statistical analysis of language," *Journal of Expert Systems with Applications*, vol. 40, pp. 2992–3000, 2013.

[19] J. P. Dickerson, V. Kagan, and V. Subrahmanian, "Using sentiment to detect bots on Twitter: Are humans more opinionated than bots?" in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2014, pp. 620–627.

[20] C. Teljstedt, M. Rosell, and F. Johansson, "A semi-automatic approach for labeling large amounts of automated and non-automated social media user accounts," in *Proceedings of the Second European Network Intelligence Conference (ENIC 2015)*, Karlskrona, Sweden, 2015, pp. 155–159.

[21] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, 1948.

[22] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Proceedings of the Tenth National Conference on Artificial Intelligence*, ser. AAAI'92. AAAI Press, 1992, pp. 129–134.

[23] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data." *BMC Bioinformatics*, vol. 10, 2009.

[24] K. J. Archer and R. V. Kimes, "Empirical characterization of random forest variable importance measures," *Comput. Stat. Data Anal.*, vol. 52, no. 4, pp. 2249–2260, Jan. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.csda.2007.08.015

[25] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1010933404324