Identifying Correlated Bots in Twitter

Nikan Chavoshi(✉), Hossein Hamooni, and Abdullah Mueen

University of New Mexico, Albuquerque, USA
chavoshi@unm.edu

Abstract. We develop a technique to identify abnormally correlated user accounts in Twitter, which are very unlikely to be human operated. This new approach of bot detection considers cross-correlating user activities and requires no labeled data, as opposed to existing bot detection techniques that consider users independently, and require large amount of recently labeled data. Our system uses a lag-sensitive hashing technique and a warping-invariant correlation measure to quickly organize the user accounts in clusters of abnormally correlated accounts. Our method is 94 % precise and detects unique bots that other methods cannot detect. Our system produces daily reports on bots at a rate of several hundred bots per day. The reports are available online for further analysis.

1 Introduction

Automated accounts, called bots, are common in social media. Although all bots are not bad, bots are easy means to engage in unethical and illegal activities in social media. Examples of such activities include selling accounts [18], spamming inappropriate content [1], and participating in sponsored activities [7]. Many social metrics are calculated based on social media data [3,15]. The significant presence of bots in social media will make many of these metrics useless. The exact number of bots is dynamic and unknown. The range of the estimates is between 3 % [18] to 7 % [14]. Social media sites, such as Twitter, regularly suspend abusive bots [19]. Yet, the number of bots is growing because of almost zero-cost in creating new bots.

Existing bot detection methods are not capable of fighting such evolving set of bots. There are several reasons. Current methods are mostly non-adaptive, require supervised training, and consider accounts independently [6,20]. Typical features used in some of the methods need a long duration of activities (e.g. weeks) [21] which makes the detection process useless, as the bots can initiate a fair amount of harm before being detected. Moreover, bots are becoming smarter. They mimic humans to avoid being detected and suspended, and increase throughput by creating many accounts. We take a novel *unsupervised* approach of *cross-correlating* account activities, that can detect such dynamic bots as soon as two hours after starting their activities.

Our *novelty* is in using activity correlation as an absolute indicator of bot behavior. Millions of users interact in social media at any time. Even at this large scale, human users are not expected to have highly correlated activities in social

E. Spiro and Y.-Y. Ahn (Eds.): SocInfo 2016, Part II, LNCS 10047, pp. 14–21, 2016.
DOI: 10.1007/978-3-319-47874-6_2

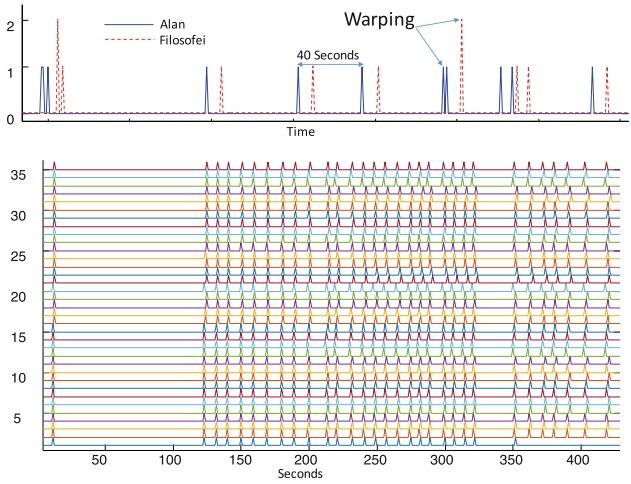

Fig. 1. (top) Two highly correlated activity sequence (six minutes) of two Twitter users: Alan and Filosofei. Warping-invariant correlation between them is 0.99, while cross-correlation is 0.72 and Pearson's correlation is 0.07. (bottom) A group of 35 correlated bots' activity sequence. Note that slight misalignment in the timestamps.

media for hours. However, in Twitter, large groups of such correlated user accounts are actively operating. A video capture of two completely unrelated (no one follows the other) and yet perfectly correlated Twitter accounts is shown in [2]. You can find several examples of activity sequence of correlated Twitter accounts in Fig. 1 (better in high resolution). Such correlation in tweeting activities is only possible if the accounts are controlled automatically, indicating that the accounts are bots. We provide mathematical significance of our approach and empirically achieve 94 % precision of our approach.

In the rest of the paper, we discuss the level of significance of correlated bots and show empirical evaluation. We omit technical details of our method, named DeBot, due to limited presentation scope. The daily reports of bots and an expanded paper are available in [2].

2 Significance of Correlation in Bot Detection

In this section, we analyze the significance of correlation in detecting bots. We first assume each user tweets independently and then relax the restriction.

We estimate the probability of two users having n posts at identical timestamps among m seconds when there are N such active users. We assume the users are independently tweeting. There are $M = m^n$ possible ways a user can post n actions in m seconds. Let us estimate the probability $\hat{p}$ that no two users have n identical timestamps under user independence.

$$\hat{p} = 1 \times \frac{M-1}{M} \times \frac{M-2}{M} \times \ldots \times \frac{M-N+1}{M}$$

The probability p of at least two users posting at the same n seconds in m seconds is simply $1 - \hat{p}$.

$$p = 1 - \frac{M!}{M^N (M-N)!}$$

Note that, if $N > M$ then $p = 1$, as there are more trials (i.e. users) than possible options (i.e. combination of seconds). If we realistically set $N = 10^9$ and $m = 3600$, p sharply goes down from one to zero, when we move from $n = 6$ to $n = 8$. Therefore, observing two users with seven or more identical posting timestamps is an extremely unlikely event when users are independent.

Let us now consider the warped instance of the above estimation. If the warping constraint is w, then we can pessimistically assume that any pair of the n tweets are more than $2w$ apart. This ensures that, for each of the n tweets, there can be a maximum of $W = 2w + 1$ locations available for an equivalent tweet. The new expression for $\hat{p}$ is the following.

$$\hat{p} = 1 \times \frac{M - W^n}{M} \times \frac{M - 2W^n}{M} \times \ldots \times \frac{M - NW^n}{M}$$

Similar to the exact matching, in case of warped matching, $p = 1 - \hat{p}$ tends to zero for $n = 13$ when $w = 20$ seconds, $N = 10^9$ and $m = 3600$.

Let us now consider the dependent case where the Twitter users react to similar news/events in similar ways. Let us assume q is the probability of a user reacting to any tweet within $\pm w$ seconds of the relevant tweet. The probability of none of the n tweets of a user fall within $\pm w$ of n tweets from another user is $1 - q^n$. The expression for $\hat{p}$ becomes the following.

$$\hat{p} = 1 \times (1 - q^n) \times (1 - 2q^n) \times \ldots \times (1 - Nq^n)$$

Note that, in the equal probability case, $q = \frac{2w}{m}$, which is identical to the $\hat{p}$ for warped correlation. In an extreme scenario, if users are perfectly in sync, $q = 1$ ensures $\hat{p} = 0$ and $p = 1$. If $q = 0.25$, p tends to zero for $n = 40$ and if $q = 0.5$, p tends to zero for $n = 80$. However, $q = 0.25$ is an extremely high probability. To elaborate, consider how many tweets/posts, that a user sees, is retweeted or shared. For an average user, it may be one in every few. Now consider how many a user shares within w seconds of *seeing*, which should be much less. Then consider how many a user shares within w seconds of another user *authoring* the tweets or retweets, which should be even smaller.

Thus, even for this unlikely high probability of a user tweeting or retweeting within ± 20 seconds ($q = 0.25$) of another tweet, the probability of two users with forty or more matching tweets in an hour is close to zero. Our system, therefore, considers users with at least forty tweets in an hour and identifies highly correlated (≥ 0.995) users as bots because of their extreme unlikelihood of being humans. This approach of identifying bots is highly precise with almost *no false-positive*.

One may think that evading detection by this simple approach is a very easy task. It is indeed very simple to evade such detection by inserting unbounded random time delays among the same tweet from many accounts. However, such randomization will severely damage the throughput of a bot-master, making it worthless to maintain large pool of uncontrolled bots. Moreover, although evasion is fairly easy, we have detected hundreds of thousands of unique correlated bots that are freely operating in absence of such a simple detection system.

We do not claim that correlated bot detection is the solution to bot related problems in social media. Detecting benign or malicious bot is out of the scope of this work. We simply suggest that detecting correlated bots has a potential to improve the performance of suspension systems that safeguard large social networks, eventually increasing the cost of bot operation and maintenance.

A pathological argument against correlated bot detection is that a human user may be identified as bot if some bots *mimic* the human user. If a human user is mimicked by bots, it is an urgent matter to take some action, such as blocking all of the accounts and asking all the users to prove their humanity once again. Naturally, only the human user can prove it while the bot mimickers will just remain blocked.

3 Empirical Evaluation

As per the discussion in the previous section, synchronized behavior in a sequence of forty activities is a near absolute indicator of automated accounts. In this section, we show empirical evaluation in comparison to other bot detection approaches.

We calculate relative support from other methods to the bots detected by our system. We compare against five methods. We have run bot discovery in every 4 hours for sixteen days (May 18 - June 3, 2015) and merged all the clusters into one consolidated set of clusters using *friend-of-friend* approach. We picked the top ten clusters in size that contained a total of 9,134 bot accounts to form our **base set** to compare against other methods.

- We compare the support to our method by Twitter's suspension process. We first ask the question, *how many bots that we detect are later suspended by Twitter?* If Twitter suspends them, we are certain that the bots were bad ones. On June 12, 2015, we began tracking these accounts via Twitter API to check whether or not they were suspended. We checked every few days until August 28, 2015. Twitter increasingly suspended more bots that we had detected months ahead. Twitter suspended 2,491 accounts in the very first probe and reached to 4,126 in the last probe. This means that roughly **45 %** of the bots were suspended by Twitter in 12 weeks.
- A successful existing technique developed in the Truthy project [6] is *Bot or Not?*, which is a supervised technique to estimate the probability of an account being bot. It uses account features, network features and content features to train a model [6] and estimates a probability of *"being bot"* for a

given account. We set a threshold of 50 % or more to classify an account as bot and found that **59 %** of the bots in our *base set* were also flagged by *Bot or Not?* on June 12, 2015. We probed *Bot or Not?* for the *base set* two more times and noticed no significant change in detection performance.

– We compare our method to an existing per-user method [21] which uses the dependence between minute-of-an-hour and second-of-a-minute as an indicator for bot accounts. The method in [21] tests the independence of these two quantities using the χ^2 test and declares an account bot if there is any dependence. The method fails for user *alan26oficial* (the same Alan as in Fig. 1) because of independence among the quantities, while our method can detect *alan26official* because of its correlation with *FrasesFiIosofos* (the same Filosofei in Fig. 1). We calculate the relative support from the χ^2 test method and identify **76 %** of the bots are supported by the χ^2 test.
– We evaluate the bots using contextual information such as tweet content and cross-user features. We investigate whether the synchronously aligned tweets have identical texts and authors. We define the *"botness"* of a group of accounts as the average of the botness of all the pairs of accounts in the cluster. For a given pair, botness is the percentage of aligned tweets that also match in their content (e.g. author, text). The higher the botness score the more successful DeBot is. We achieve an average of **78 %** botness when we match text and/or authors of the tweets. Simply put, the aligned tweets have identical text and authors **78 %** of the time. Note that there is a very little difference between **and** and **or** configuration. This suggests that most of the time tweets and authors match.
  Less botness score does not necessarily mean that our method is detecting false positives. We see many bot accounts that correlate in time perfectly, but do not have identical tweets.
– We investigate if approximate text matching would increase botness by employing human judges in Amazon Mechanical Turk. We ask the judges to determine whether fifty random pairs of accounts are showing similar text (may not be exact), URLs, authors and languages. We then calculate the botness. DeBot achieves up to 94 % botness score from the contextual information. Simply put, **94 %** of the tweets are not only synchronized in time, but also share the same information (Table 1).

Table 1. Relative support of different tests of DeBot

	Twitter	BotOrNot?	χ^2 Test	Text &Author	Text || Author	Human Judgment
Relative Support	45 %	59 %	76 %	78 %	79 %	94 %

3.1 Recall Estimates

It is impossible to calculate the exact recall of a bot detection technique because a complete list of known bots does not exist. we estimate the recall of three

bot detection methods by a simple approach. First, we listen to the Twitter streaming API for 30 min and pick those user that have more than 1 activity to be able to calculate DTW distances. In 30 min we filter out 8600 user accounts, on average. We test these accounts using *Bot or Not?* and χ^2 test methods. We apply DeBot to identify the bots based on temporal correlation.

The final results, which are the average of three rounds of our experiments, show the highest recall rate of 6.3 % for DeBot, which is very close to the true bot ratio (8.5 %) estimated and disclosed by Twitter recently [17]. *Bot or Not?* achieves 3.4 % bot detection rate.

4 Related Work

Real-time correlation monitoring has been a well-researched topic for over a decade now. One of the first works is StatStream [22], which can monitor thousands of signals. In [16], authors show a method to monitor lagged correlation in streaming fashion for thousands of signals. In [5], authors develop a sketch (i.e. random projection) based correlation monitoring algorithm that does not consider time warping. Twitter stream can provide tweets of millions of users which are at least an order of magnitude more in number, and an order of magnitude less in density than the method in [5], and time warping exists in Twitter. Such warped sparseness has not been addressed previously for correlation monitoring.

A good characterization of spammers in Twitter is presented in [10]. Authors concluded that 92 % of the accounts that Twitter suspends for spamming activities are suspended within three days of the first post. Therefore, if a spamming bot survives one week, it is very likely to survive a long time. Our work identifies bots that are tweeting for months, if not years. In [18], authors characterize the spam detection strategies very well. Spam detection methods that analyze social graph properties, characterize contents and rates of postings, and identify common spam redirect paths, are typically *at-abuse* methods. Such methods find the spam after the spam has done the harm. In contrast, our method can detect accounts registered by account merchants which will eventually be sold to miscreants, and thus, our method detects these bots *soon-after-registration* to prevent future abuse. *Detecting bots by correlating users is our novelty.*

Other relevant works include detecting campaign promoters in Twitter [12]. Correlating user activity across sites (e.g. Yelp and Twitter) can provide useful information about linked-accounts, and thus, form a basis of privacy attack [9]. In [8], authors perform offline analysis to discover *link-farming* by which spammers acquire a large number of followers. In [13], authors develop a fast algorithm to mine millions of co-evolving signals and find anomalies. In [4], authors find temporally coherent collaborative Liking of Facebook pages. The authors in [11], present a method to characterize groups of malicious users. They consider three features such as individual information, and social relationships to provide deep understanding of these groups. As opposed to most of these works, our focus is to correlate within the same site to identify bot accounts that already are or will potentially become spammers.

5 Conclusion

We introduce a real-time method that detects bots by correlating their activities. Our method can detect hundreds of bot accounts everyday, which now have aggregated to hundreds of thousands of bots in eight months. Human judges in Amazon Mechanical Turk have found the detected bots are highly similar to each other. Our method, DeBot, is identifying bots at a higher rate than the rate Twitter is suspending them. In comparison to per-user methods, our cross-user temporal method detects more bots with strong significance.

References

1. How twitter bots fool you into thinking they are real people. http://www.fastcompany.com/3031500/how-twitter-bots-fool-you-into-thinking//-they-are-real-people
2. Supporting web page containing video, data, code and daily report. http://www.cs.unm.edu/~chavoshi/debot/
3. Asur, S., Huberman, B.A.: Predicting the future with social media. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, vol. 1, pp. 492–499. IEEE, Aug. 2010
4. Beutel, A., Xu, W., Guruswami, V., Palow, C., Faloutsos, C.: Copycatch: stopping group attacks by spotting lockstep behavior in social networks. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 119–130. International World Wide Web Conferences Steering Committee (2013)
5. Cole, R., Shasha, D., Zhao, X.: Fast window correlations over uncooperative time series. In: Proceeding of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining - KDD 2005, p. 743 (2005)
6. Ferrara, E., Varol, O., Davis, C., Menczer, F., Flammini, A.: The rise of social bots. arXiv preprint 2014. arXiv:1407.5225
7. Galán-García, P., de la Puerta, J.G., Gómez, C.L., Santos, I., Bringas, P.G.: Supervised machine learning for the detection of troll profiles in twitter social network: application to a real case of cyberbullying. In: Herrero, Á., et al. (eds.) International Joint Conference SOCO'13-CISIS'13-ICEUTE'13. AISC, vol. 239, pp. 419–428. Springer, Heidelberg (2014)
8. Ghosh, S., Viswanath, B., Kooti, F., Sharma, N.K., Korlam, G., Benevenuto, F., Ganguly, N., Gummadi, K.P.: Understanding and combating link farming in the twitter social network. In: Proceedings of the 21st International Conference on World Wide Web - WWW 2012, p. 61. ACM Press, New York, April 2012
9. Goga, O., Lei, H., Parthasarathi, S.H.K., Friedland, G., Sommer, R., Teixeira, R.: Exploiting innocuous activity for correlating users across sites, pp. 447–458, May 2013
10. Grier, C., Thomas, K., Paxson, V., Zhang, M.: @spam: the underground on 140 characters or less. In: Proceedings of the 17th ACM Conference on Computer and Communications Security - CCS 2010, p. 27. ACM Press, New York, October 2010
11. Jiang, J., Shan, Z.-F., Wang, X., Zhang, L., Dai, Y.-F.: Understanding sybil groups in the wild. J. Comput. Sci. Technol. **30**(6), 1344–1357 (2015)
12. Li, H., Mukherjee, A., Liu, B., Kornfield, R., Emery, S.: Detecting campaign promoters on twitter using markov random fields. In: 2014 IEEE International Conference on Data Mining (ICDM), pp. 290–299 (2014)

13. Matsubara, Y., Sakurai, Y., Ueda, N., Yoshikawa, M.: Fast and exact monitoring of co-evolving data streams. In: 2014 IEEE International Conference on Data Mining, pp. 390–399. IEEE, December 2014
14. Morstatter, F., Carley, K.M., Liu, H.: Bot detection in social media: networks, behavior, and evaluation. In: ASONAM - Tutorial, August 2015
15. Ruiz, E.J., Hristidis, V., Castillo, C., Gionis, A., Jaimes, A.: Correlating financial time series with micro-blogging activity. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining - WSDM 2012, p. 513. ACM Press, New York, February 2012
16. Sakurai, Y., Papadimitriou, S., Faloutsos, C.: Braid: Stream mining through group lag correlations. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, p. 610 (2005)
17. Subrahmanian, V., Azaria, A., Durst, S., Kagan, V., Galstyan, A., Lerman, K., Zhu, L., Ferrara, E., Flammini, A., Menczer, F., Waltzman, R., Stevens, A., Dekhtyar, A., Gao, S., Hogg, T., Kooti, F., Liu, Y., Varol, O., Shiralkar, P., Vydiswaran, V., Mei, Q., Huang, T.: The darpa twitter bot challenge. IEEE Comput. **1**, 38–46 (2016). (In press)
18. Thomas, K., Paxson, V., Mccoy, D., Grier, C.: Trafficking fraudulent accounts: the role of the underground market in twitter spam and abuse trafficking fraudulent accounts. In: USENIX Security Symposium, pp. 195–210 (2013)
19. Twitter. The Twitter Rules. https://support.twitter.com/articles/18311
20. Wang, A.H.: Detecting spam bots in online social networking sites: a machine learning approach. In: Foresti, S., Jajodia, S. (eds.) DBSec 2010. LNCS, vol. 6166, pp. 335–342. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13739-6_25
21. Zhang, C.M., Paxson, V.: Detecting and analyzing automated activity on twitter. In: Spring, N., Riley, G.F. (eds.) PAM 2011. LNCS, vol. 6579, pp. 102–111. Springer, Heidelberg (2011). doi:10.1007/978-3-642-19260-9_11
22. Zhu, Y., Shasha, D.: StatStream: statistical monitoring of thousands of data streams in real time. In: Proceedings of the 28th International Conference on Very Large Data Bases, volume 54 of VLDB 2002, pp. 358–369 (2002)

http://www.springer.com/978-3-319-47873-9

Social Informatics
8th International Conference, SocInfo 2016, Bellevue, WA, USA, November 11-14, 2016, Proceedings, Part II
Spiro, E.; Ahn, Y.-Y. (Eds.)
2016, XIX, 517 p. 122 illus., Softcover
ISBN: 978-3-319-47873-9