Temporal Patterns in Bot Activities

Nikan Chavoshi
University of New Mexico
Albuquerque, NM 87131
chavoshi@unm.edu

Hossein Hamooni
University of New Mexico
Albuquerque, NM 87131
hamooni@unm.edu

Abdullah Mueen
University of New Mexico
Albuquerque, NM 87131
mueen@unm.edu

ABSTRACT

Correlated or synchronized bots commonly exist in social media sites such as Twitter. Bots work towards gaining human followers, participating in campaigns, and engaging in unethical activities such as spamming and false click generation. In this paper, we perform temporal pattern mining on bot activities in Twitter. We discover motifs (repeating behavior), discords (anomalous behavior), joins, bursts and dynamic clusters in activities of Twitter bots, and explain the significance of these temporal patterns in gaining competitive advantage over humans. Our analysis identifies a small set of indicators that separates bots from humans with high precision.

CCS Concepts

•**Information systems** → **Social networks;** *Web mining;*

Keywords

Time series mining; Pattern mining; Bot; Social Media

1. INTRODUCTION

Social media sites attract bot masters to create and maintain large number of bots, i.e. automated user accounts. Bots are used to harvest human followers, run advertising, election, marketing campaigns and spread unethical content. One of the studies has shown that 8.5% of accounts in Twitter are bots [12]. Twitter strictly monitors these automated accounts and suspends them regularly, however, the number of bots is still increasing [5] making them interesting to study.

In this paper, we consider extracting temporal patterns from activity time series of thousands of Twitter bots. Activity time series record the number of Twitter activities such as tweet, retweet and delete done by a user at every millisecond. Note that there can be more than one activity in the same millisecond because of network and server delay.

We apply state-of-the-art time series pattern mining algorithms on activity time series such as motif discovery [10], discord discovery [14], subsequence join [8] and dynamic clusters [9]. We have discovered each of these patterns in Twitter bot activities and interpreted them. We develop a set of four features to evaluate the results of our bot detection platform, called *DeBot* [5, 4] . These features show that accounts labeled bot by DeBot are completely different from human accounts and classifying bots from humans based on them gives us 81% precision.

We first describe DeBot and discuss our data collection process in Section 2. We describe the temporal patterns in Section 3 and a comparison between human and bot accounts based on the learning from the patterns in Section 4. We conclude in Section 5.

WWW 2017,April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145/3041021.3051114

2. DEBOT ARCHIVE

DeBot is a live bot detector that identifies unusually synchronous Twitter accounts. DeBot has been running since August 2015 and found 700K unique bots so far (February 2017). DeBot detects 1500 unique bots on average everyday by processing about a million tweets generated by several thousands Twitter accounts.

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

n = number of sequential synchronous posts in an hour
q = probability of acting upon seeing another activity
N = Number of active users, we assume 1B

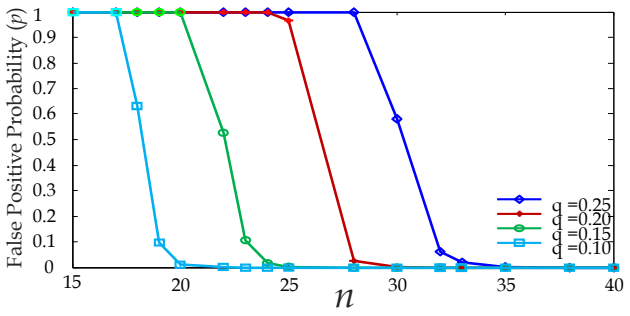

Figure 1: The probability of false positive is almost zero when users are *highly* correlated (>0.99) for *large* number of activities (>40).

DeBot works based on a simple principle: if a group of users is highly synchronous for sufficiently long time, the group cannot be of humans. This is a surprisingly simple technique that does not require language dependent features, long history of activity, and human supervision in labeling training data. Figure 1 shows the probability of false positives with respect to the length of sequentially synchronous activities. In other words, how many sequential

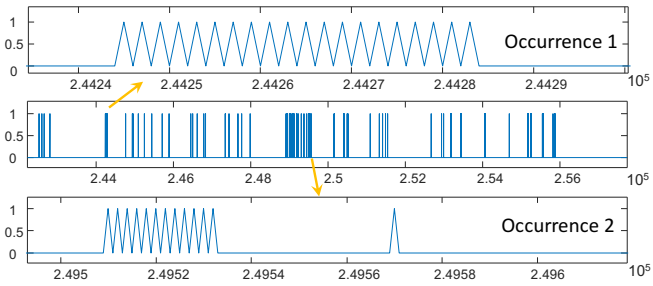

Figure 2: Example of time series motif in bot activities. x-axis is in millisecond, y-axis shows number of tweets.

correlated activities from two users are sufficient to make sure that they are bots. Assuming one billion independent active users, we see 40 synchronous tweets in one hour is significant enough to declare two accounts as bots.

We refer readers to a YouTube video [1] that shows screen captures of such accounts. Readers can also find more detailed information on how the formula is derived in [5] which analyzes the significance of correlation in detecting bots. It is true that bot detection through unusual synchronicity is easily avoidable, however, DeBot is detecting more than thousands of new bots everyday for over a year (08/2015-02/2017), which implies that there was no action to suspend these bots during this time.

We analyze a set of bot groups/clusters to understand whether they are semantically associated. We show some of the clusters and the *names* of the accounts in Table 1. Several of these Twitter handles/names may be suspended at the time of reading, or may not belong to original accounts anymore. The table shows numerous bot groups that are semantically similar within their groups. For example, the Racing cluster is mostly related to Australia and the News cluster mostly contains celebrity news accounts to catch mass attention. The clusters also show *content* similarity as detected by the Mechanical Turk users. For example, the Serial accounts mostly post tweets in Asian languages including Thai, Korean and Arabic.

One may also want to see more breakdowns on other features to know how bot accounts are distributed based on their languages, countries, genders, ages and etc. Since most of bot accounts do not provide a valid demographic information in their profile, doing statical analysis on these features yields inaccurate results that are not reliable. Although there may be some reliable pieces of information such as latitude and longitude provided by third party application, the number of bots using these applications is not significant.

In general, DeBot successfully finds groups of bots that are related in their names, languages, and locations just by analyzing their temporal behavior. Temporal analysis lets DeBot be language independent because it does not need to consider the context of tweets. We will exploit features of bots to understand the motivation and production process of the bot masters in the future. To analyze temporal patterns in bot activities, we select a set of 1500 bots from DeBot and record their activities over ten continuous days. Detected bots and their activity time series are available at our project's web page [2].

[1]https://youtu.be/1YFhbBsZ8zs

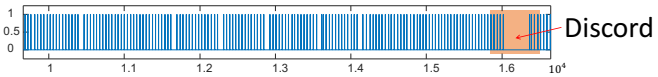

Figure 3: Example of discord in bot activities. x-axis is in millisecond, y-axis shows number of tweets.

3. TEMPORAL PATTERNS

We apply five temporal pattern mining algorithms on the bot activity series and describe several successful cases in this section.

3.1 Motif Discovery

Time series motif is a repeating subsequence in a long time series [10]. Motif can be very simply defined as the most similar pair of subsequence. Motif discovery is an important data mining tool to identify preserved structure in the underlying dynamics of the data source. We use a time warping distance measure, Awarp [11], to extract the most similar repeated segments for each bot.

In Figure 2, we show the activity time series of the user `@DSGuarico` for five days. Visually there is no periodicity in the activity other than some long pauses. However, the user has a motif that occurs many times (two occurrences are shown in Figure 2). The motif is simply a sequence of tweets made at about 500 milliseconds interval (exact interval varies). Clearly it is not possible for a human being to post a tweet at this rate even if the tweets are identical. Upon further investigation, we observe that all of these tweets are copied from the President of Venezuela, Nicolás Maduro. `@DSGuarico` was synchronous with at least fifty other bots engaged in similar kind of proliferation of political tweets.

3.2 Discord Discovery

Time series discord is the most anomalous subsequence in a long periodic time series. Discord is defined as the subsequence whose nearest neighbor is the furthest among other nearest neighbors. A good segment of Twitter bots are periodic. For example, `@countforever` is a harmless bot that just counts periodically. Another example is `@RedSwingline1` which posts political content periodically. A discord in such bots is unusual and potentially indicates downtime in the system that controls the bot.

In Figure 3, we show the bot `@m_and_e_2` that is periodically posting every 4 seconds. We discover a discord of 32 seconds long pause.

3.3 Periodicity

Periodicity detection is a common pattern mining tool to identify repeated behavior. We consider finding the most frequent periodicity in our set of bots. We evaluate periodicity by considering the most frequent delay between successive activity. Figure 4 shows the distribution where three frequent periodicity dominate others. Half minute, two minutes and seven to eight minutes of periodicity are commonly observed.

There are some bots that produce tweets and retweets at a high rate in a short period, such as the bot shown in Figure 3. These bots mainly retweet arbitrary content from the network to remain active and to increase their chance to gain human followers.

Cluster Name	Examples
Serial accounts	2jo116, 2jo120, 2jo24, 2jo31, 2jo42, 2jo64, 2jo72, 2jo88, 2jo97, 2jo_71
News	ADavisNews, ARiversNews, AlYankovicTNews, BilbaoAllNews, ChemtrailsTNews, ChromeAllNews, DYankeeNews, PaulinaNoticias, ShakiraNoticia1
Racing	AusHorseRacingN, AusRacingTweets, CanterburyRacin, FreeRaceTips_, FreshRacing_, HorseRacingAus1, RacingAussie_, RacingFields, RacingTweetsAU
Japanese	AzamiMisaki, KaguraKokona, KawakamiAyumu, KisaragiMinami, KizekiEfy, akataniHaruna, engyo_bot, gutarajunko, guzuguzu6, i_san, nonkina_tousan, ochame_p, tekitohiroko, yontanbot
Indian	AadarshSvebpvme, BhatNipun, BinduSing, DaluiNityananda, LullaAbhishek, RoyRoymukul, SinghKulvira, YoVinaykumar, abhishekbhsker, anil_khar, arvindtomar_, baloni_sunil, desh_raj_, euzvfsdtxud, mohitsharma_1, rajeshkumara_, sahilver_, sumit_vai, sumitkumarsha, sushilkumr_, vikram_nag
Mobile	MobileStandared, m_plusplus4, miconmob, mob_charger, mob_maps, mobilesmrt, mobileupdate1, product_mobile, attack_mobile, boss_mobileboss, m_authorize, miconmob, mobile_external, mobilefollower3, mobilefuture2, mobilelearning7, mobilesmrt, mobilesubscb, mobmuseums, mobrepeat, mob_design, mob_hole
Love	Awkward_Loves, Awkwardlovetext, BaeLoveNotes, Funnyloves012, HistoryTabloids, ItsLoveLetter, Lovelythink1, LovequQuite, LovesQuote0, LovingFacz, Truelovesfacts, girlfriendloved, justlovforever, loveQuoites, love_fillings, lovelikefuny, lovemsgs512, lovenoteguru, loveromantic60, lovingfaczzz, lovingsrose, points_love
Random	102f2kid, 10_vivid, 125gn3a6, 142d4afaf, 17E4a3fb, 18Hghhgjgjd, 19Kytghgfdd, 1oTalalaykina, 229ae, 22Tjdtjtgkytk, 25Gfhthysjtj, 26Gjghtrhysxrf, 28_ghjtrjtyjtgh, 2Asagao543210, 2Ic65ec, 2ch33n5, 2gbm8p7, 2hgddg, 2j8p3ab, 33634m87, 37Hkyjdtytyhjgh, 3Vistlip, 3bf72, 3en2p, 5Asagao543210, 5Mbityutskikh, 6Asagao543210, 7759c5, 79_shamilya, 7Asagao543210, 9F6m36
East European	2016Kuramshina, 4uOkaderkaev, 9_chitanova90, Bravkov73K, ERodygina, IrodionVozilkin, Izoldiya87, KoveraV, MrPoveteva, NiceMelkov, Q5Esarumova, SunnyZavoloka, SuperNoyanov, VladianaLuchuk, ZZbogumanova, ZariyaZhuchenko, cenyceCla,chekannikova73, chekannikovnik1, dennawarneckem1, erogozova1, fsoltanovich, nice_sosedka, nlataev, nnekhoroshkova, penerova1, polunichevalip1,

Table 1: Example clusters of synchronous Twitter accounts with common naming. Find more at [2].

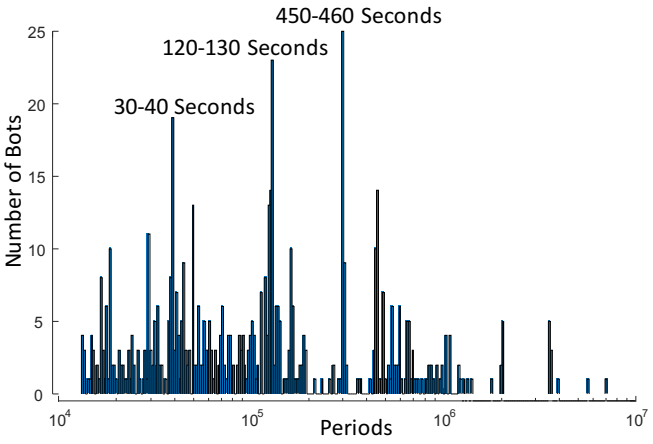

Figure 4: Distribution of periodicity

3.4 Time Series Join

Time Series Join on subsequences identifies segments of two time series that are very similar at an arbitrary lag. As DeBot detects synchronous groups of bots, the bots in the same group have long join sequences. However, bots in different groups have no particular reason to have a join sequence.

We perform join between every pair of time series from different correlated groups. We discover pairs of bots that are overall uncorrelated but contain highly correlated join sequences. In Figure 5, we show the total activities of two accounts for 18 hours. It is clear that the total activities of these two users are not synchronous. However, if we zoom in on the segment in which both users have synchronicity, we find that there is no tweet or retweet in these segments. Both users were deleting tweets that they posted previously.

The above observation is a novel one as no previous bot detection system considered deletion activity as a feature. Further investigation of these two accounts in Figure 5 reveals that they both strongly support a political party in Turkey named the Justice and Development Party (AKP). During the General Election in Turkey in 2015, the AKP allegedly hired thousands of trolls to create a strong online presence [3]. We hypothesize that the trolls use multiple accounts to do their activities on Twitter, and use automated tools to delete the tweets to maintain an average profile. Unfortunately, Twitter does not provide the text of the deleted tweets, hence, we do not know if they were recycling tweets or just maintaining a low total number of tweets.

We also observe that the deletion of a large number of tweets is a common bot behavior. Bots try to have the same *net content generation rate* as benign accounts. A benign account creates 5.1 and deletes 0.7 tweets on average in two hours, so $5.1 - 0.7 = 4.4$ tweets are accumulated every 2 hours. Bots detected by DeBot, also show identical increase

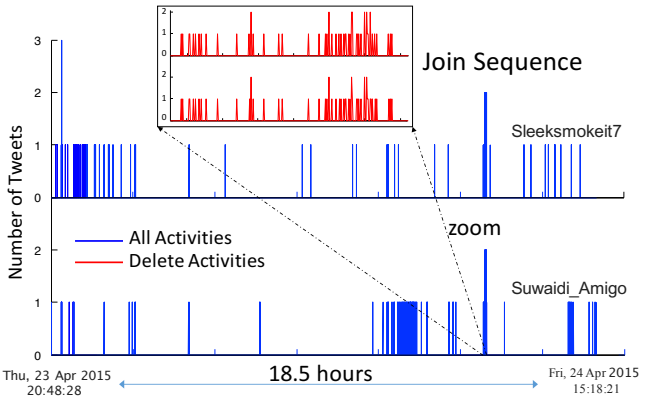

Figure 5: Total activities of two users over 18 hours show no correlation. A zoomed in segment of deletion activities show perfect correlation suggesting the accounts are bots.

in accumulated content in two hours ($13.8 - 9.4 = 4.4$. See Figure 10). Thus, many bots maintain a low profile closer to normal users by deleting the previous tweets.

3.4.1 *Profiling Deletions*

We set to profile the deletion activities to understand the general approach bots are taking. We take a small subset of 1600 bot accounts from DeBot archive randomly. We listen to the activities of these accounts for 2 hours. For each user, we look at the total number of deletions and the most frequent interval between two successive deletions. We plot 550 users with more than 10 deletions in these 2 hours in Figure 6.

We see two clear clusters in the figure. The top cluster consists of user accounts that delete frequently in every 600 seconds (i.e. 10 minutes). The bottom cluster has no specific periodicity, and the most frequent interval is 50 seconds or less. When the number of deletions is less than or around 100, there is no periodicity and no burst, as shown for the user 4. Accounts with high numbers of deletions either show strong periodicity such as users 1 and 2 in Figure 6, or show bursty behavior like user 3. Note that user 3 deletes up to 8 tweets in a second which is an unrealistic rate of activities for a human to perform.

3.5 Dynamic Clusters of Bots

We would like to know if bots change their groups dynamically. Unfortunately, Twitter provides a partial view on the users. It is possible to have both overlapping and disjoint clusters while they all belong to one global cluster. Therefore, we simplify our task to discover if a single bot changes cluster membership by changing its activity pattern.

We have found examples in which three accounts, A,B and C are related. Initially A and B were correlated, and later A moves out of B's group and joins C's group. One example is given in Figure 7, captured by tracking three bots for 24 hours.

Dynamic changes in bot activity is not well understood. The exact algorithms or dynamics are not known. However, the motivation behind all such changes is to behave as humans to avoid suspension and attract followers.

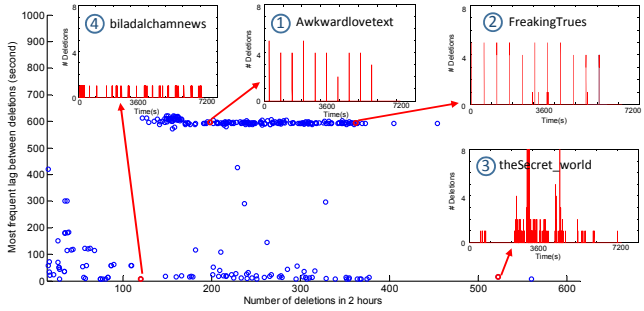

Figure 6: Deletion behavior of bot accounts. Two clear clusters exist, one that deletes every 600 seconds (i.e. 10 minutes) and the other that has bursty deletion behavior with no periodicity.

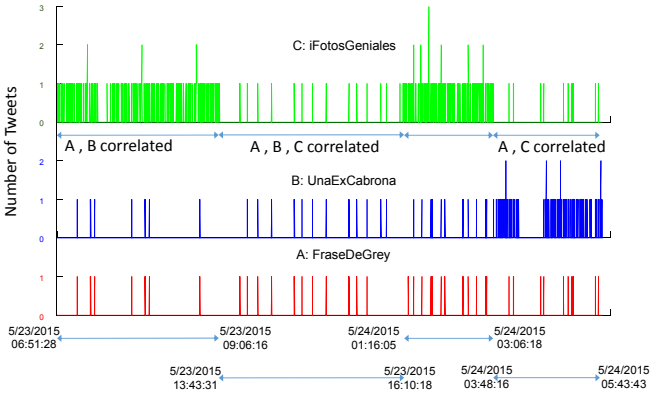

Figure 7: Three users A, B and C changing their groups.

3.5.1 *Correlated Handovers*

We correlate infrequent Twitter actions with bot activities to demonstrate that bots are not only synchronous in tweets, retweets and deletes, they are also synchronous in changing their Twitter screen-name (handle). Twitter accounts are allowed to change their screen-names at any time. [7] shows that handing over a screen-name is a common behavior among suspicious accounts in Twitter.

In Figure 8, we show three bots that are synchronous (with 0.96 correlation) for over 11 weeks. We also point to the times when the bots changed their URLs and some other accounts picked up those URLs shortly. We see the bots perform URL handovers within the same hour. The motifs are shown in Figure 8. The URLs that were handed over by these accounts are all related to celebrities such as MacMiller, Rihanna, Drake, Megan Fox and Lil Wayne.

The above explanation provides an evidence that bots work in correlation, possibly using the same code-base, and that they hand over at the same time to swap or pass URLs that they do not want to lose. In the future, we will investigate how to scale handover detection in real time so we can track the interest areas of the bots and take countermeasures.

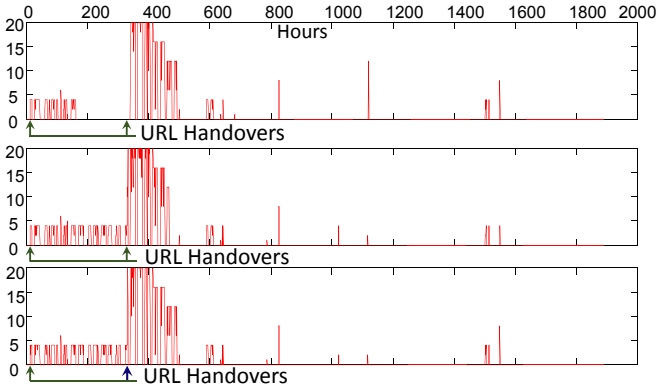

Figure 8: Three accounts with almost identical activity profile and correlated handovers. Handovers initiate change in activity patterns.

3.6 Burst in Bots

Although we get over 1500 bots on average every day, Figure 9 shows that there are several days when number of detected bots is significantly higher. Figure 9 shows that there are two consecutive days on April 2016 when ∼12,000 accounts were flagged as bots. Most of automated accounts of these two days were supporting three popular music bands (One Direction, 5 Seconds of Summer, and 5 Harmony). On those two days, there was a music award show organized by *iHeartRadio* which had an award category called *Best Fan Army*. Fans had to vote for their favorite singer or band online. Most bots we detected on those days had a hashtag related to one of these bands or *Best Fan Army* award. There can be two conclusion: 1) The fans or bands hired a group of bots to propagate information about the award and make the band name one of the top trends in Twitter, and/or 2) These bots might have been used for online mass voting to manipulate the result of the contest.

4. COMPARISON TO HUMANS

This section is designed to show the difference between bot accounts and real accounts. In the first step we identify a set of 7000 accounts that are not found suspicious by DeBot and Bot or Not? [1], and are not suspended by Twitter. We name them **benign** users. In the second step we define four high risk indicators:

- The number of activities per user in two hours is a generic feature focusing on overall activities. Bots are usually very active.
- The number of deletions per user in two hours indicates whether or not the user maintains a low profile on the accumulated number of tweets to avoid looking like a bot. Similar to overall activities, bots delete tweets more frequently than benign accounts.
- The percentage of tweets that contain URLs indicates what fraction of the contents of the tweets are potentially outside of Twitter.
- The percentage of the duplicate tweets [13] indicates the fraction of the tweets which is generated by the user automatically. We consider all the tweets with identical text as duplicates. This set includes the retweets by definition. The original sources of these duplicate contents are usually celebrities, politicians, sportsmen and news accounts.

A high value in any of the above indicators is a sign of abnormal behavior. We compare the above indicators of the benign accounts and of the bots detected by DeBot, Twitter and *Bot or Not?*. We run our bot detection algorithm 50 times to correctly estimate the variance of the indicators in the sets of benign and bot users. The results shown in Figure 10 clearly separates bots and benign users.

To properly estimate the predictive power of the above high risk indicators, we perform 10-fold cross validation using a Support Vector Machine (SVM) classifier with an average accuracy of 81.71% on a balanced set of benign and bot users. We use an RBF kernel with $\sigma = 1$. The confusion matrix of the classifier is shown in Table 4.

One may ask the question that why we did not use these indicators in the first place instead of DeBot to identify bot accounts. The answer is that this experiment is possible because DeBot has already labeled a set of accounts for us. If we did not have the labeled data, we would not be able to do the comparison and classification based on these features. So, this section is not proposing a new approach to detect bot accounts. It shows how much bots and benign accounts labeled by DeBot are different.

	Classified Benign	Classifier Bot
True Benign	65%	35%
True Bot	11.2%	88.8%

Table 2: Confusion Matrix

4.1 Comparison to Non-temporal Methods

To test DeBot contextually with Twitter and *Bot or Not?*, we collected the activities of the bot accounts that DeBot detected for two weeks and calculate the above indicators. After two weeks, we identify the accounts that are suspended by Twitter and the accounts that have more than 50% probability of being a bot in *Bot or Not?*. The indicators for the three sets of accounts are presented in Figure 11. In this experiment, we processed 7 million tweets in total to observe the following:

- The three bot detection algorithms tend to agree on the percentage of tweets that contain URLs and the percentage of duplicate tweets.
- DeBot catches high deletion activities more than others while Twitter catches high overall activity more than others.
- The benign users have the smallest values for all of the indicators. This is a very significant difference between the bots detected by the three methods and the benign users.

5. CONCLUSION

Most existing work considers almost all different aspects of bots such as textual and network aspects [6]. Temporal aspects of bots has been largely ignored. In this paper, we perform temporal pattern mining on bot activity time series, and observe interesting behavior of Twitter bots. Our future work will explore bot dynamics in a systematic way to decode the underlying mechanisms of these bots.

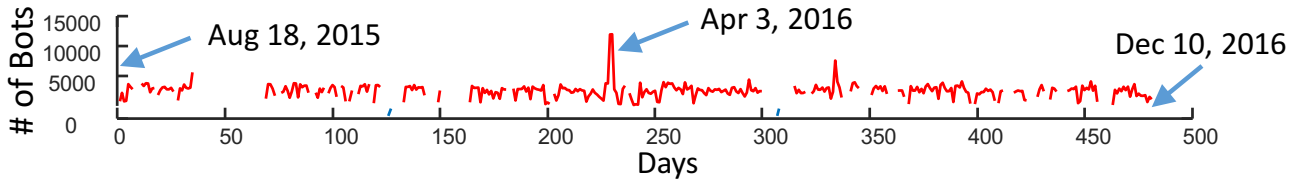

Figure 9: Number of bots detected by DeBot per day. Gaps indicate downtime due to update and maintenance.

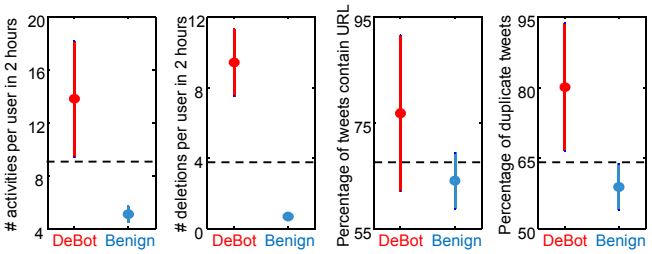

Figure 10: Comparison between benign accounts and accounts we (*DeBot*) detect as bot. The dashed lines show near complete separation in all of the features between benign accounts and the accounts *DeBot* detected considering the mean and the standard deviation.

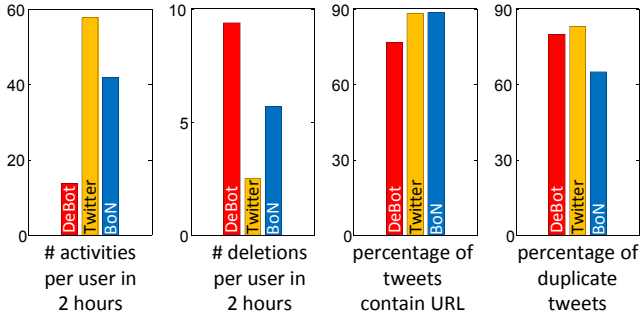

Figure 11: Comparison between accounts we detected as bot (*DeBot*), accounts suspended by Twitter and accounts detected as bot by Bot or Not? (*BoN*)

6. REFERENCES

[1] Bot or not? a truthy project. http://truthy.indiana.edu/botornot/.

[2] Supporting web page containing video, data, code and daily report. www.cs.unm.edu/~chavoshi/debot.

[3] Targeted journalists react as ak party trolls hint at new operation. http://www.todayszaman.com/anasayfa_targeted-journalists-react-as-ak-party/-trolls-hint-at-new-operation_354568.html.

[4] N. Chavoshi, H. Hamooni, and A. Mueen. DeBot: Twitter Bot Detection via Warped Correlation. In *Proceedings of the IEEE International Conference on Data Mining*, ICDM '16, 2016.

[5] N. Chavoshi, H. Hamooni, and A. Mueen. Identifying correlated bots in twitter. In *Social Informatics - 8th International Conference, SocInfo 2016, Bellevue, WA, USA, November 11-14, 2016, Proceedings, Part II*, pages 14–21, 2016.

[6] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 6 2016.

[7] H. Hamooni, N. Chavoshi, and A. Mueen. On URL changes and handovers in social media. In *Social Informatics - 8th International Conference, SocInfo 2016, Bellevue, WA, USA, November 11-14, 2016, Proceedings, Part I*, pages 58–74, 2016.

[8] H. Hamooni and A. Mueen. Dual-domain Hierarchical Classification of Phonetic Time Series. In *ICDM 2014*, ICDM, 2014.

[9] A. Mueen, H. Hamooni, and T. Estrada. Time Series Join on Subsequence Correlation. In *2014 IEEE International Conference on Data Mining*, pages 450–459. IEEE, 12 2014.

[10] A. Mueen, E. Keogh, Q. Zhu, S. Cash, and B. Westover. Exact Discovery of Time Series Motifs. *Proceedings of the 2009 SIAM International Conference on Data Mining*, pages 473–484, 2009.

[11] C. N. Mueen Abdullah, N. Abu-El-Rub, H. Hamooni, and A. Minnich. Fast Warping Distance for Sparse Time Series. In *Proceedings of the IEEE International Conference on Data Mining*, ICDM '16, 2016.

[12] V. S. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, F. Menczer, R. Waltzman, A. Stevens, A. Dekhtyar, S. Gao, T. Hogg, F. Kooti, Y. Liu, O. Varol, P. Shiralkar, V. Vydiswaran, Q. Mei, and T. Huang. The DARPA Twitter Bot Challenge. Jan. 2016.

[13] K. Tao, F. Abel, C. Hauff, G.-J. Houben, and U. Gadiraju. Groundhog Day: Near-duplicate Detection on Twitter. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1273–1284, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[14] D. Yankov, E. Keogh, J. Medina, B. Chiu, and V. Zordan. Detecting time series motifs under uniform scaling. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining KDD 07*, KDD '07, page 844, 2007.