Kritya Gorowara, 2021, 9:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

Artificial Neural Network Based Social BOT Detection

M.Tech. Scholar Kritya Gorowara, Asst. Prof. Jayshree Boaddh
Department of Computer Science and Engineering,
Mittal Institute of Technology,
Bhopal

Asst. Prof. Jashwant Samar
Department of Computer Science and Engineering,
UIT RGPV, Bhopal

Abstract- Digital platform dependency of today era attract promoters to brand product services. So unwanted posting was done by some programs known as bot. Number of researchers have proposed different techniques to identify these bots which was post by bot programs. This paper has developed a model to identify bots from real user. User action were analyze as features for classification of bots and real user. Whole process adopt graph based clustering and teacher learning based optimization genetic algorithm. Graph based clustering classify user into two class and genetic algorithm find the class representative action sequence in form of features. Experiment was done on real twitter dataset and result shows that proposed model has increase the detection accuracy of work.

Keywords:- Clustering, Data mining, Genetic Algorithm, Social Network. Online Social Networks (OSNs), Twitter, Botmers, Legitimate users.

I. INTRODUCTION

By using the Internet, it has become quite natural to receive any type of information from any source around the world. The increased demand from social sites allows users to gather an abundance of user information and data. Enormous amounts of data on these pages often draw the attention of fake users [1]. Twitter has rapidly become an online source for acquiring real-time information about users. Twitter is an Online Social Network (OSN) where users can share anything and everything, such as news, opinions, and even their moods.

Several arguments can be held over different topics, such as politics, current affairs, and important events. When a user tweets something, it is instantly conveyed to his/her followers, allowing them to outspread the received information at a much broader level [2]. With the evolution of OSNs, the need to study and analyze users' behaviors in online social platforms has intensity. Many people who do not have much information regarding the OSNs can easily be tricked by the fraudsters. There is also a demand to combat and place a control on the detection of robot (BOT) in social networking sites attracted the attention of researchers. BOT detection is a difficult task in maintaining the security of social networks. It is essential to recognize bots in the OSN sites to save users from various kinds of malicious attacks and to preserve their security and privacy.

These hazardous maneuvers adopted by bots cause massive destruction of the community in the real world. Twitter bots have various objectives, such as spreading invalid information, fake news, rumors, and spontaneous messages. Bots achieve their malicious objectives through advertisements and several other means where they support different mailing lists and subsequently dispatch bot messages randomly to broadcast their interests. These activities cause disturbance to the original users who are known as non-bots. In addition, it also decreases the repute of the OSN platforms.

Several research works have been carried out in the domain of Twitter bot detection. To encompass the existing state-of the- art, a few surveys have also been carried out on fake user identification from Twitter. Tingmin et al. [4] provide a survey of new

© 2021 Kritya Gorowara. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

methods and techniques to identify Twitter spam detection. The above survey presents a comparative study of the current approaches. On the other hand, the authors in [5] conducted a survey on different behaviors exhibited by bot/ spammer on Twitter social network. The study also provides a literature review that recognizes the existence of spammer on Twitter social network.

Despite all the existing studies, there is still a gap in the existing literature. Therefore, to bridge the gap, we review state-of-the-art in the bot detection and fake user identification on Twitter. Moreover, this survey presents a taxonomy of the Twitter bot detection approaches and attempts to offer a detailed description of recent developments in the domain.

II. RELATED WORK

Ameen and Kaya [6] proposed out a related work and found that easygoing backwoods had the greatest accomplishment at 92.95%. An examiner must research to discover the best calculation to use before going with further examination. There is no fastidious calculation that goes past all others under all conditions; this explains the need of exploration with various methodologies. Prior to moving towards higher classifier techniques, it is important to value the reason that most of analysts have release SVM classifiers, for example, sack of-words and pack of-implies.

Lee et. al. [7] deployed social honeypots consisting of genuine profiles that detected suspicious users and its bot collected evidence of the spam by crawling the profile of the user sending the unwanted friend requests and hyperlinks in MySpace and Twitter.

Features of profiles like their posting behaviour, content and friend information to develop a machine learning classifier have been used for identifying spammers. After analysis profiles of users who sent unsolicited friend requests to these social honey pots in MySpace and Twitter have been collected.

LIBSVM classifier has been used for identification of spammers. One good point in the approach is that it has been validated on two different combinations of dataset – once with 10% spammers+90% non-spammers and again with 10% non-spammers+90% spammers. Limitation of the approach is that less dataset has been used for validation.

Viswanath et al. [8] discover that dependency on community detection makes more vulnerable to Sybil attacks where honest identities conform strong communities. Because Sybils can infiltrate honest communities by carefully targeting honest accounts. That is, Sybils can be hidden as just another community on OSN by setting up a small number of the targeted links. The targeted links are the links given to the community which contains the trusted node.

They make an experiment by allowing Sybils to place their links closer to the trusted node instead of random nodes, where closeness is defined by ranking used by the community detection algorithm they employ. Hence, Sybil nodes are high ranked in the defence scheme. Naturally, it leads to Sybils being less likely to be detected for that attack model because Sybils are appeared as part of the local community of the trusted node.

Boshmaf et al. [9] point out that structure-based Sybil detection algorithms should be designed to find local community structures around known honest (non-Sybil) identities, while incrementally tracking changes in the network by adding or deleting some nodes and edges dynamically in some period for better detection performance.

Alshehri et al. [10] use hashtags and N-grams to show out grown-up Arabic substance. The pack of-words procedure uses twofold qualities to guarantee for positive words in a posted substance, while sack of-implies include discovering a normal of word vectors. The result of their inspect was a 79% precision of preparing.

Peining Shi, et. al in [11] novel method of detecting malicious social bots, including both features selection based on the transition probability of clickstream sequences and semi-supervised clustering, is presented in this paper. This method not only analyzes transition probability of user behavior click streams but also considers the time feature of behavior.

International Journal of Science, Engineering and Technology

An Open Access Journal

III. PROPOSED METHODOLOGY

Explanation of proposed model BDNN (BOT Detection by Neural Network) was done in this section by flow chart of figure 1.

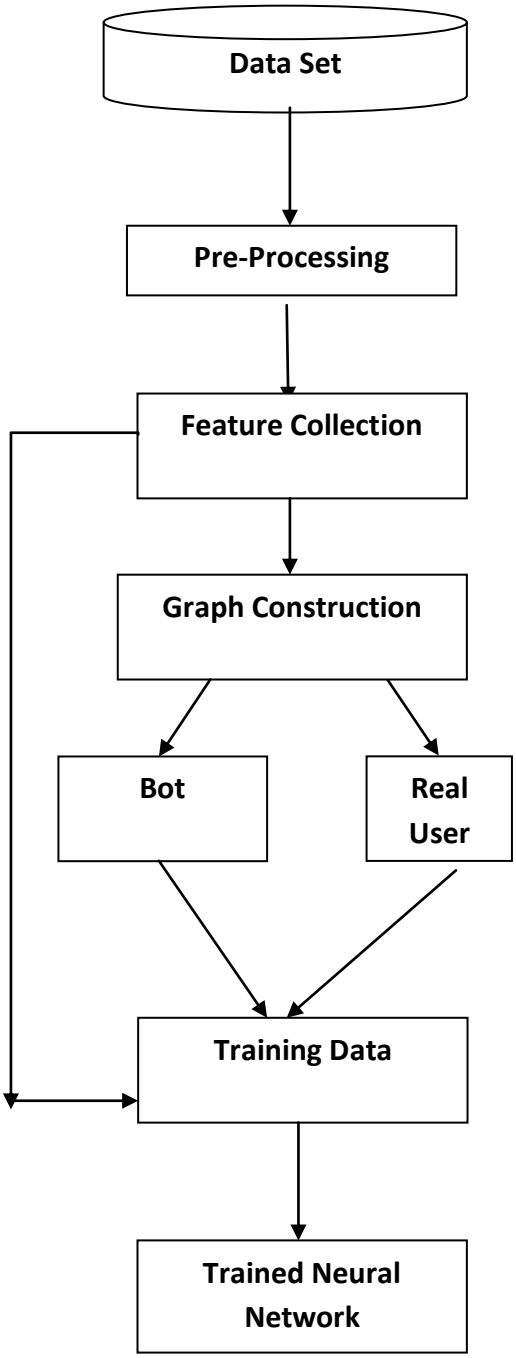

Fig. 1 Proposed work Block diagram.

1. Pre-Processing:

As the dataset is a collection of data which is unorganized and need to retrieve important information which is fruitful for the work in this work dataset contain time, date, protocol, session, etc. Here data is clean and transform this as per working environment. Preprocessing is a procedure utilized for transformation of content into feature vector.

As tweet content on webpage have words which need pre-processing by removing stopwords [14]. So set of stopwords are removed and filtered words were further process to collect keywords. Hence each tweet has its own set of keywords depend on type of content. Although common keywords may exist between users of same domain.

So let tweet Tm have content {w1, w2, s1, w3, s2, s1…….wn} where n is total number of words in Tm page. After stopword {s1, s2,…..} removal important words will be {w1, w2, w3,…….wn} [12].

2. Feature collection:

This work twitter dataset was consider as the input where five features of users were extract. These features F represent the user behavior on the social network. List of features are:

- Sequence of Shares
- Sequence of Likes instance
- Sequence of Tweets perform by user
- Sequence of re-tweet perform by user
- Time instance of the event

Feature from 1 to 5 can be easily extract from the dataset where as per the behavior follow by the user.

Transition probability between action events steams: P(i,j) represents the probability that the click action is j at timestamp t followed by click action at timesptamp t-1. So probability find the relation between the i and j features of the use in form of transition done in a specific set of durations.

$$P(i,j) = \frac{\sum_1^t X_i \rightarrow X_j}{\sum_1^t X_i}$$

In above equation $X_i \rightarrow X_j$ act as transition form i to j feature instance, while t as the total time instance when i feature were applied. So if work use n number of features than each user has a feature vector of nxn where cell contain a probability transition value.

$$\begin{vmatrix} P_{11} & P_{12} & P_{13} \end{vmatrix}$$

$$F_x = \begin{vmatrix} P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{vmatrix}$$

3. Graph Construction:

In this step develop a completely connected graph where each node is connect with other node and distance between them act as weight of the edge. Estimation of the distance was done by using X and Y axis of the system. Here this can be understand as let nodes are N = {n_1, n_2, n_3,.....n_m} and distance between them are evaluate by Euclidian distance formula.

Now sort graph edges with Minimum Weight in a decreasing order. This can be understand as matrix S[] of three column and rows depend on number of edges present in the graph.

4. Resultant cluster:

So nodes which are present with less distance edge weights are considered as the true user or real user of the social media. While nodes whose distance values are larger in oter cluster were consider as the bots.

As each bot set of instance sequence were totally different from real user set of instances, so distance from other existing nodes were high. Hence cluster selection of real or bot is depends on the weight value of the partial tree present in the cluster.

5. Training of Error Back Propagation Neural Network:

- Let us assume a three-layer neural network.
- Now consider i as the input layer of the network while j is considered as the hidden layer of the network.
- Finally, k is considered as the output layer of the network.
- If w_{ij} represents a weight of the between nodes of different consecutive layers.
- So the output of the neural network depends on the below equation:

$$X_j = \sum x_i \, . w_{ij} b_j$$

Where, $1 \leq i \leq n$; n is the number of inputs to node j, and b_j is the biasing for node j.

Once the system gets output than it gets to compares with the desired watermark value in this case instead of 0 or 1 work has assumed the 0 or 100 value output. Hence the network will learn the weights between layers and constant threshold range from 0 to 100. This error needs to be correct by adjusting the weight values of each layer by eq. 8 to 12, [35].

So estimation of error was done by Eq. 8.

$$\frac{\partial E_i}{\partial O_i} = \frac{\partial(-1 * ((y_i * \log(O_i) + (1 - y_i) * \log(1 - O_i)}{\partial O_i}$$

$$\frac{\partial E_i}{\partial o_i} = (-1 * ((y_i * \log(O_i) + (1 - y_i) * \log(1 - O_i) \text{----Eq. 8}$$

Similarly, other values can be calculated to find another set of derivatives using the above equation. For each input to neuron calculate the derivative concerning each weight using equation. Now let us look at the final derivative by Eq. 9.

$$\sum_{i=1:n} \frac{\partial H_i}{\partial W_{i(j,k)}} = \frac{\partial(h_{i(ouput)} * W_{i(j,k)})}{\partial W_{i(j,k)}} - Eq. 9$$

Now by using the chain rule, final derivates were calculated for the below equation. Here multiplication of each derivative was done in eq. 10

$$\frac{\partial E_i}{\partial W_i} = \frac{\partial E_i}{\partial O_i} * \frac{\partial O_i}{\partial H_i} * \frac{\partial H_i}{\partial W_i} - - - Eq. 10$$

So overall ∂W_i can be obtained by getting the value of weight from the above equation, here all set of weight which need to be update are change by eq. 11 values.

$$\partial W_i = \begin{bmatrix} \frac{\partial E_1}{\partial W_{1,1}} & \frac{\partial E_2}{\partial W_{1,2}} & \frac{\partial E_3}{\partial W_{1,3}} \\ \frac{\partial E_1}{\partial W_{2,1}} & \frac{\partial E_2}{\partial W_{2,2}} & \frac{\partial E_3}{\partial W_{2,3}} \\ \frac{\partial E_1}{\partial W_{3,1}} & \frac{\partial E_2}{\partial W_{3,2}} & \frac{\partial E_3}{\partial W_{3,3}} \end{bmatrix} - - - - Eq. 11$$

The ANN weight updates were done by the above matrix of ∂W_i.

$$W_{ij} = W_{ij} + \partial W_{ij} \text{----Eq. 12}$$

So end of the above iteration steps over when error obtained from the output layer get nearer to zero or some constant such as 0.0001.

6. Final Solution:

In this work after sufficient number of epochs neural network learns the neural network. This trained neural network will assign users class to Bot, Real user cluster as per transition probability feature values.

IV. EXPERIMENTAL SETUP

Whole work was implement on MATLAB software. It is utilize on account of its rich library which has numerous inbuilt storage that can be specifically use in this work for various reason.

Out of various storage few are crossing point, contrasting of the string, and so forth. One more essential factor is its GUI by which one who doesn't know about the code can straightforwardly runs the storage without having earlier information.

1. Dataset:

In this work experiment is done on social dataset content obtained from https://botometer.iun i.iu.edu/bot-repository/datasets.html, where as per the user related twitter comments of respected user with different action and timestamp were available.

2. Results:

Results of proposed model was compared with previous work proposed in [14].

Table 1. Accuracy Based Comparison.

Data Size	Proposed Work	Previous Work
20000	0.4545	90
24000	47.37	68.42
28000	0.5652	69.57
32000	62.96	74.07
36000	65.52	75.86

Above table 1 has shown that proposed model has increase the bot detection accuracy of work. User probability of next activity increase the clustering accuracy of the work. Paper has learned the feature values of clustered users and increased the work accuracy for user.

Table 2. Precision Based Comparison.

Data Size	Proposed Work	Previous Work
20000	0.5556	1
24000	0.6923	1
28000	0.7647	0.9412
32000	0.8095	0.9524
36000	0.8261	0.9565

Above table 2 has shown that proposed model has increase the bot detection precision parameter as compared to previous work. It was found that user trail behavior on the social site were transform in probability of transition increases the precion parameter. As training data for neural network was processed by graph based clustering algorithm.

Table 3. Recall Based Comparison.

Data Size	Proposed Work	Previous Work
20000	0.7143	0.8182
24000	0.6	0.6842
28000	0.6842	0.7273
32000	0.7391	0.7692
36000	0.76	0.7857

Above table 3 has shown that proposed model has increase the bot detection recall of work. User probability of next activity increase the clustering accuracy of the work. Paper has learned the feature values of clustered users and increased the work recall for user.

Table 4. F-measure Based Comparison.

Data Size	Proposed Work	Previous Work
20000	0.625	0.8182
24000	0.6429	0.8125
28000	0.7222	0.8205
32000	0.7727	0.8511
36000	0.7917	0.8627

Above table 4 has shown that proposed model has increase the bot detection F-measure parameter as compared to previous work. It was found that user trail behavior on the social site were transform in probability of transition increases the F-measure parameter. As training data for neural network was processed by graph based clustering algorithm.

V. CONCLUSIONS

Life of social media depends on real user action but digital user perform unfair action and reduce overall trust value. Many of social site execute bot detection algorithm. This paper has proposed a graph based algorithm for clustering of social uer into bot and real cluster.

Input of this algorithm was set of user action and based on these action transition probability features user were cluster into two class. Output of graph based clustering algorithm were user feature value wth a tag of bot or real. This labeled dataset was used for training of neural network. Experiment was done on real twitter user dataset and result shows that proposed model has improved the accuracy of the model by 40.21%, while f-measure was improved by 14.65%. In future scholar can adopt other machine learning model to increase the accuracy of work.

REFERENCES

[1] Mevada D. L., Daxini V., "An opinion bot analyzer for product Reviews using supervised machine Learning method." pp.03, (2015).

[2] M. N. Istiaq Ahsan , Tamzid Nahian , Abdullah All Kafi, Md. Ismail Hossain, Faisal Muhammad Shah "Review Bot Detection using Active Learning." 978-1-5090-0996-1, pp.16, (2016).

[3] Michael C., et al. "Survey of review bot detection using machine learning techniques." Journal of Big Data 2.1, pp.9, (2015).

[4] Adike R. G., Reddy V,. "Detection of Fake Review and Brand Bot Using Data Mining Technique.", pp.02,(2016).

[5] Rajamohana S. P, Umamaheswari K., Dharani M., Vedackshya R., "Survey of review bot detection using machine learning techniques.", 978-1-50905778-8, pp.17 (2017).

[6] Ameen, A.K.; Kaya, B. Detecting botmers in twitter network. Int. J. Appl. Math. Electron. Comput. 2017, 5, 71–75.

[7] Kyumin Lee, James Caverlee, Steve Webb, Uncovering Social Spammers: Social Honeypots + Machine Learning, Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval, 2010, Pages 435–442, ACM, New York (2010).

[8] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove, "An analysis of social network-based sybil defenses," ACM SIGCOMM Computer Communication Review, vol. 40, pp. 363-374, 2010.

[9] Y. Boshmaf, K. Beznosov, and M. Ripeanu, "Graph-based sybil detection in social and information systems," in Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on, 2013, pp. 466-473.

[10] Alshehri, A.; Nagoudi, A.; Hassan, A.; Abdul-Mageed, M. Think before your click: Data and models for adult content in arabic twitter. In Proceedings of the 2nd Text Analytics for Cybersecurity and Online Safety (TA-COS-2018), 2018.

[11] Peining Shi, Zhiyong Zhang And Kim-Kwang Raymond Choo. "Detecting Malicious Social Bots Based on clickstream Sequences". IEEE Access March 18, 2019.

[12] Mubarak, H.; Darwish, K.; Magdy, W. Abusive language detection on Arabic social media. In Proceedings of the FirstWorkshop on Abusive Language Online, Vancouver, BC, Canada, 4–7 August 2017; pp. 52–56.