Introducting Bot Secure, Twitter Based Bot Detection Solution Powered By Machine Learning

Praveen Kumar S[1], Chandran M K[2], Kumthan S[3], Rubiga Sri S[4]
[1]Assistant Professor, Dept of CSE
[2, 3, 4]Dept of CSE
[1, 2, 3, 4]Excel Engineering College Komarapalayam, Tamil Nadu

Abstract- *All the social networking sites which allow the users to express their opinion on various topics like politics, sports, stock market, entertainment etc. It is one of the fastest means of conveying information. It highly influences people's perspective. So it is necessary that tweets are sent by genuine users and not by bots. A bot sends spam messages. Therefore detecting of bots helps to identify spam messages. This paper proposes an approach to detect the twitter bots using machine learning algorithms. We compare Decision tree, Multinomial Naïve Bayes, Random Forest and Bag of Words.*

Keywords- Web technology, Bot

I. INTRODUCTION

Twitter is one of the most rapidly growing social networking sites. It allows users to share news, talk about their opinions and discuss current affairs. The users can follow people with similar interests or opinions. The users can instantly send tweets to their followers. The information can reach larger people by the means of re-tweeting. The tweets increase spontaneously during live events like sports or award shows. Twitter can be accessed using smartphones or computers. Paid promotions can be done resulting in large revenue generation and also to increase the sales of the products. Twitter enables students to gain additional information about the topics taught in the class. The message which is shared with the followers is called tweet. The tweet should be concise and can have a maximum length of 140 characters. Hashtag (#) is used to find and follow a particular topic. When a hashtag is popular it becomes a trending topic. The links in twitter are bi directional; where user may have follows and followers. In Twitter if you follow someone then you can see all their tweets if the account is public, this doesn't mean that he/she can see your tweets.

Users receive many tweets in which some of them are from bots. Detecting bots is necessary to identify the fake users and to protect the genuine users from misinformation and malicious intents. Twitter bot is software that sends tweets automatically to users. Bots are designed for doing activities such as spamming. The malicious intent of Twitter bots are: 1) to spread rumors and false news. 2) To defame someone's character. 3) False communications are created to steal credentials. 4) Users are misdirected towards fake web sites. 5) To change thoughts of an individual or group example influencing popularity.

We are using dataset from Kaggle. It contains attributes like number of followers, friends, location, screen name (used to communicate online), verified (if the user is authenticated), favorite (used for liked tweets), url, id, description, listed count. Features are extracted based on spearman correlation coefficient. The data set is trained to identify bots. We are implementing Decision Tree, Multinomial Naïve Bayes, Random Forest and Bag of words. The algorithm with highest accuracy is used to test real time data.

II. LITERATURE SURVEY

Several works have been done in Twitter Bot Detection. The methods and work done is presented below: Fake identities created by humans or bots are detected using machine learning models which are dependent on engineered features. It was evaluated whether readily available and engineered features that are used for the successful detection, using machine learning models, of fake identities created by bots or computers can be used to detect fake identities created by humans. Supervised machine learning algorithms require a dataset of features with a label classifying each row or outcome. Features are thus the input used by supervised machine learning models to predict an outcome. These features can be the attributes found via APIs that describes a single piece of information about an SMP account, like the number of friends. The predictive results from the trained machine learning models only yielded a best F1 score of 49.75%. The machine learning models were trained to use engineered features without relying on behavioral data [1]. Content polluters, or bots that hijack a conversation for political or advertising purposes are a known problem for event prediction, election forecasting and when distinguishing real news from fake news in social media data. Identifying this type of bot is particularly challenging. Content polluters are bots that attempt to subvert a genuine discussion by hijacking it for political or advertising purposes. Methods were

developed to identify social bots in data using only partial information about the user and their tweet history, in real time. They investigated two characteristics of tweets i.e. temporal information and message diversity. It was found that content polluters in this dataset often timed their tweets together. By analysing the temporal patterns one could infer the presence of bot accounts. It was also found that bots used a small set of URLs in their tweets [2].

III. PROPOSED SYSTEM AND METHODOLOGY

The block diagram of our system is shown in figure 1 and 2

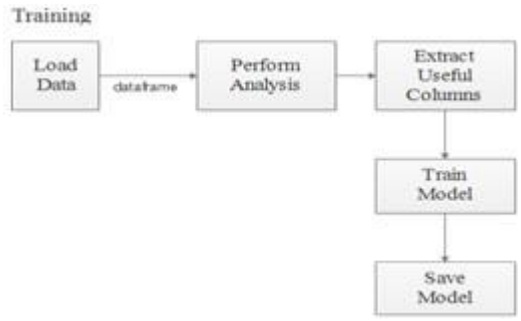

Figure 1: Training of the dataset

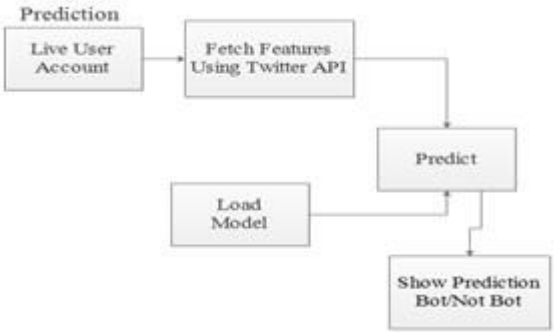

Figure 2: Prediction of Real-time data

The train data has many attributes. The required features are extracted using Spearman correlation method. Three learning models are built namely, Naïve Bayes algorithm, Decision Tree, Random Forest and Bag of words. The best learning model is applied on real-time data as shown in figure 1 and 2. The data is preprocessed and null values are removed using pandas (tool for preprocessing). The dataset is trained and the testing dataset is the real-time data on Twitter. The output is in the form of 0 or 1(1 indicating that it is a bot and 0 indicating that it is not a bot)

IMPLEMENTATION

This gives a brief description about implementation details of the system in terms of algorithms. Four algorithms are implemented namely, Decision Tree, Multinomial Naïve Bayes, Random Forest and Bag of Words.

Decision Tree

Algorithm 1 shows the implementation details of Decision Tree algorithm. Require: detects bots based on number of followers, friends, screen name, description, location and verified #tag

Ensure: Converts the above feature into binary values.

1) Whole training set is considered as root.
2) Information gain is used to choose which attribute to label each node with
3) Recursively construct each subtree on the training instance that would be classified down the path of the tree.
4) If all positive or negative instances remain label that node “yes” or “no” accordingly
5) If no attributes remain, label with majority vote is left at that node

If no instances remain, label with a majority vote of the parent’s training instance.

Multinomial Naive Bayes

Algorithm 2 shows the implementation details for Multinomial Naive Bayes algorithm. Require: System detects bots based on number of followers, friends, screen name, description, and location and verified.

Ensure: Converts the above feature into binary values. Hypothesis is the given account is bot.

1) P(h|d) is the probability of hypothesis h given the data d.
2) P(d|h) is the probability of data d given that the hypothesis h was true.
3) P(h) is the probability of hypothesis h being true.
4) P(d) is the probability of the data (regardless of the hypothesis).
5) P(h|d)=(P(d|h)*P(h)/P(d) return (P(hjd)).

Random Forest

Algorithm 3 shows the implementation details for detecting bots using Random Forest algorithm. Require: System detects bots based on number of followers, friends, screen name, description, and location and verified. Ensure: Converts the above feature into binary values.

1) Randomly select “k” features in given m features
2) Among “k” features calculate node d using best spilt point
3) Spilt the node into daughter nodes using best split point
4) Repeat 1 to 3 until “l” number of nodes has reached
5) Forest is built by repeating steps 1 to 4 “n” times to create “n” number of tress to give random forest
6) Use each tree on the test feature and store outcome
7) Calculate votes for each predicted outcome
8) Consider highest voted outcome as the final prediction

Bag of Words

Algorithm 4 shows the implementation details for detecting bots using Bag of Words algorithm. Require: detects bots based on number of followers, friends, screen name, description, and location and verified

Ensure: Converts the above feature into binary values
Data is collected about the known bot accounts
Vocabulary is designed. It consists of single word, 2 or more words. Hash representation is used.
Test data is compared with the vocabulary and stored as a binary vector
Scoring methods include counting number of times the word has appeared and frequency that each word appears in a document out of all the words in the document.

IV. RESULTS AND DISCUSSION

In ROC curves, true positive rate is plotted against false positive rate distinguishing between classes. The train dataset is split into 70% train data and 30% test data. True positive is the one that correctly identifies the true values. False positive is the one falsely identifies the true values as false.

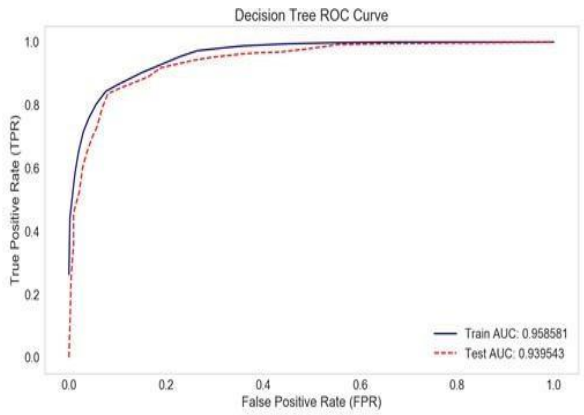

Figure 3

The accuracy for train data using Decision Tree algorithm is 88.70% and for test data is 87.85%.

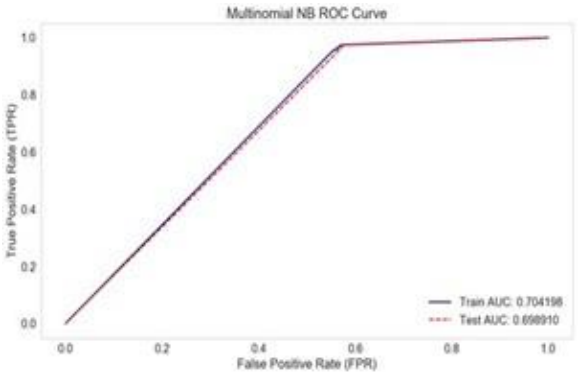

Figure 4

The accuracy for train data using Multinomial Naïve Bayes algorithm is 67.69% and for test data is 69.76%.

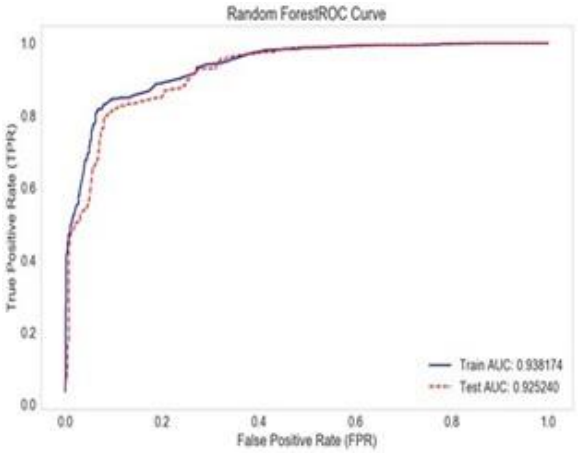

Figure 5

The accuracy for train data using Random Forest algorithm is 87.58% and for test data is 86.19%

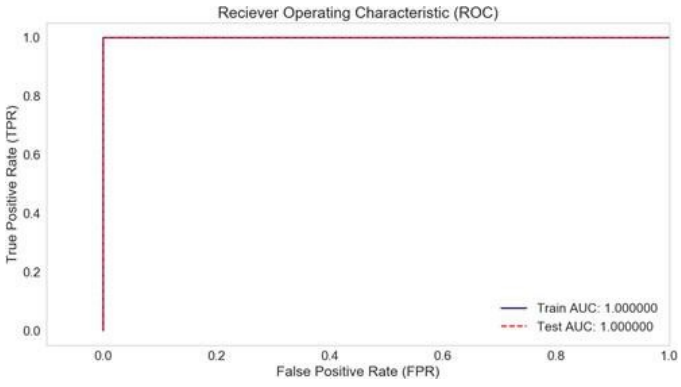

Figure 6

The accuracy for train data using Bag of words algorithm is 97.07% and for test data is 95.24%.

Table I: Performance Comparison of All Algorithms Implemented

S. No	Performance Comparison	
	Algorithm name	*Accuracy (%)*
1	Decision Tree	87.85
2	Multinomial Naïve Bayes	69.76
3	Random Forest	86.19
4	Bag of Words	95.24

The results as in table 1 show that Bag of Words performs with highest accuracy of 92% in bot detection

V. CONCLUSION

In our paper, we proposed an algorithm which detects Twitter bots. Bag of words algorithm was found to be the best learning model with an accuracy of 96.7% for train data 96.65% for test data in comparison to Decision Tree, Multinomial Naïve Bayes and Random Forest. Hence Bag of words algorithm was applied on real-time data and the Twitter bots were successfully identified.

REFERENCES

[1] Van Der Walt, Estée, and Jan Eloff. "Using machine learning to detect fake identities: bots vs humans." IEEE Access 6 (2018): 6540-6549.

[2] Sever Nasim, Mehwish, Andrew Nguyen, Nick Lothian, Robert Cope, and Lewis Mitchell. "Real-time detection of content polluters in partially observable Twitter networks." arXiv preprint arXiv:1804.01235 (2018).

[3] Khalil, Ashraf, Hassan Hajjdiab, and Nabeel Al-Qirim. "Detecting Fake Followers in Twitter: A Machine Learning Approach." International Journal of Machine Learning and Computing 7,no.6(2017).

[4] Wetstone, Jessica and Sahil R. Nayyar. “I Spot a Bot : Building a binary classifier to detect bots on Twitter.” (2017).

[5] Karataş, Arzum, and Serap Şahin. "A Review on Social Bot Detection Techniques and Research Directions." In Proc. Int. Security and Cryptology Conference Turkey, pp. 156-161. 2017.

[6] Chavoshi, Nikan, Hossein Hamooni, and Abdullah Mueen. "Identifying correlated bots in twitter." In International Conference on Social Informatics, pp. 14-21. Springer, Cham, 2016.

[7] Perdana, Rizal Setya, Tri Hadiah Muliawati, and Reddy Alexandro. "Bot spammer detection in Twitter using tweet similarity and time interval entropy." Jurnal Ilmu Komputer dan Informasi 8, no. 1 (2015): 19-25.

[8] Haustein, Stefanie, Timothy D. Bowman, Kim Holmberg, Andrew Tsou, Cassidy R. Sugimoto, and Vincent Larivière. "Tweets as impact indicators: Examining the implications of automated “bot” accounts on T witter." Journal of the Association for Information Science and Technology 67, no. 1 (2016): 232-238.