Twitter Bot Detection

# Datta Sainath Dwarampudi

Computer Science,

New York University

New York, USA

dsd298@nyu.edu

# Madhu Kiran Gudivada

Computer Science,

New York University

New York, USA

mg5309@nyu.edu

***Abstract*—** **In recent years, social media accounts are not only controlled by humans but also by bots. Recent literature has focused primarily on detection of bot in social networks. These bots act as a double-edged sword for a common social media user. Few bots generate large amount of data about news and updating feeds, while other bots spread spam or malicious data through tweets, which is of major concern. In this project, we are trying to design, analyze and implement classifiers to detect the probability of a given user account as a bot or not. We shall use 2 classifiers based on Naïve Bayes and Logistic regression for tweet analysis. We shall compare the classifiers and try to improve the better classifier till we get satisfied accuracy.**

***Keywords—machine learning, bots, Naïve Bayes, logistic regression, twitter, Natural Language Processing***

* + 1. INTRODUCTION

Our project proposal is to determine a user account as a legitimate user account or a bot. There is tremendous work been done in this domain. Bots can be used to generate live scores, weather and news. They can also be used to distribute malicious tweets which have huge consequence on society. There are. According to Emilio Ferrara, a computer scientist and assistant professor at the University of Southern California (USC) has said that online bots which influence the political discourse on social media as never. We shall employ two algorithms based on Naïve Bayes and logistic regression for tweet analysis and then hone one of the algorithm of the above two which gives a better accuracy for bot detection. We shall modify and enhance the code to get better accuracy. Online marketers on twitter are also affected due to huge number of bot in the twitter ecosystem.

* + 1. MOTIVATION

Twitter is one of the most popular social media platforms, it has been plagued by many bots during recent years. This has been a major problem to deal with. A recent publication states that more than half of the twitter accounts are not human. Few other optimistic studies states that 5-9% of the overall population is a bot and these bots generate 24% of all the tweets produced on twitter. In a case, a bot campaign created fake “buzz” about a tech company: automated stock trading algorithms acted on this chatter, resulting in a spurious 200-fold increase in market price. This has motivated us to actively deal with this issue by designing good classifiers using naïve Bayes and logistic regression for tweet analysis.

* + 1. RELATED WORK

We are referring a lot of published papers and websites to complete this project successfully. We tried to refer papers which dealt with how twitter was in the in initial phase [1],[2], to better understand about the community of social media and its usage statistics. Krishnamurthy et al. [1] has studied and segregated twitter into 3 groups: 1) broadcasters, which have a large number of followers; 2) acquaintances, which have about the same number on either followers or folling; and 3) miscreants and spammers, which follow a large number of follower but have few followers. Twitter has attracted spammers to post spam content, due to its popularity and openness. Fighting against spam on Twitter has been investigated in recent works [2],[3]&[4]. Yardi et al. [2] dated spam on twitter. His observations, include that spammer ssend more messages than legitimate users, and more likely to follow other spammers than legitimate users. Thus, a high follower-to-following ratio is a sign of spamming behavior. We are also mainly referring from papers recommended by professors [5],[6] & [7].

* + 1. DATA

We have gathered 50 bot accounts and 50 user accounts from various sources. It was very tough to gather data for bot accounts. We gathered bot data primarily from major websites which have detailed bot accounts listed in its websites. We even obtained data for bots from news website like CNN and other major websites which increased the number of followers of prominent politicians in United States of America. We tried searching for bots from various other websites, which helped people to build bots and some bot accounts were listed as the work performed by the previously. Some of the statuses contained symbols like emoticons, which were converted to UTF-8 format.

* + 1. ALGORITHM(S)

The major part of any analysis for twitter data would be the tweets. The way a bot tweets and a human tweet plays a major part in the classification of the accounts. So, for the initial part of the project would be to gather the last 200 tweets and clean them. Then these are fed to the selected classifiers.

V.I. CLEANING DATA

The data received from the twitter API is crude and contains a lot of unwanted symbols(emoticons) and hex values. We used the NLTK package to remove these symbols. Once cleaned the tweets were tokenized from which stop words are removed to improve the vocabulary of the classifier. The remaining words are stemmed to ensure that the count vectorizer identifies the similar words properly.

V.II. DATA COVERSION

The cleaned data is then labelled properly to distinguish between bot and user data. Since the algorithms require numerical data we use CountVectorizer() fuction from sklearn to covert the text data into numerical data.

# V.III TRAINING DATA

The data is spilt into Training data and Test data for a 10-fold cross validation using the StratifiedKFold technique.

V.IV. ALGORITHMS USED

One the best methods to classify text data is Naïve Bayes. As Wikipedia states that “Naïve Bayes is a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features”. For discrete features like the ones encountered in document classification (include spam filtering), Multinomial and Bernoulli distributions are very popular.

V.IV.I. Multinomial Naïve Bayes.

Wikipedia defines Multinomial Naïve Bayes as “A multinomial event model, samples (feature vectors) that represent the frequencies with which certain events have been generated by a multinomial where the probability that event occurs”. A feature vector is represented as a histogram, by counting the number of times event was observed for an instance. This is the event model typically is used for document classification, with events representing the occurrence of a word in a single document (see bag of words assumption).

V.IV. II. Bernoulli Naïve Bayes.

In the multivariate Bernoulli event model, features are independent booleans (binary variables) describing inputs. Like the multinomial model, this model is popular for document classification tasks where binary term occurrence features are used rather than term frequencies. This event model is especially popular for classifying short texts. It has the benefit of explicitly modelling the absence of terms.

A Naive Bayes classifier with a Bernoulli event model is not the same as a multinomial NB classifier with frequency counts truncated to one.

V.IV.III. Logistic Regression

We also used another popular method to classify the data i.e. by

regression. Logistic regression is defined as “a regression model where the dependent variable (DV) is categorical. This article covers the case of a binary dependent variable—that is, where it can take only two values, "0" and "1", which represent outcomes such as pass/fail, win/lose, alive/dead or healthy/sick.” Since our dependent variable has only two values i.e. bot or not bot we need not use multinomial Logistic Regression. Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative logistic distribution. Thus, it treats the same set of problems as probity regression using similar techniques, with the latter using a cumulative normal distribution curve instead.

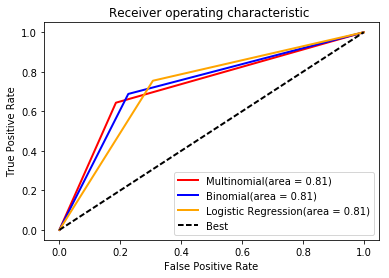
* + 1. PEFORMANCE AND RESULTS

We calculated the average accuracy, recall, precision, f1 and roc\_auc(area under curve) scores for the 10-fold values and plotted an ROC curve by using the tpr and fpr values.

The average values are in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | MultinoialNB | BernoulliNB | LogisticReg |
| accuracy | 0.7128 | 0.7227 | 0.7295 |
| precision | 0.8344 | 0.8159 | 0.7814 |
| recall | 0.6435 | 0.6878 | 0.7547 |
| f1 score | 0.7230 | 0.7436 | 0.7666 |
| roc\_auc | 0.8139 | 0.8145 | 0.8055 |

The ROC graph is:



To estimate the error rate of the classifier we used Mean Squared Error and the values are

|  |  |  |  |
| --- | --- | --- | --- |
|  | MultinoialNB | BernoulliNB | LogisticReg |
| MSE | 0.2871 | 0.2772 | 0.2704 |

VII. Future Work

The performance of the classifiers based purely on tweet data has yield average performance which can be observed from the evaluation parameters. Also, the low mean squared error on the cross-validation data is very low which shows that the classifier is having high bias which leads to underfitting of the model.

This problem has raised because of lack of complexity of the classifier. Since we are using only one parameter i.e. the tweet data for estimation the model has become biased.

So, we have decided to increase the number of parameters used for classification by add the following,

1)**Sentimental Analysis**: We would add Twitter specific sentiment analysis algorithms, including happiness, arousal-dominance-valence, and emoticon scores.

2) **Friends Hierarchy** that include an account’s social contacts, such as the median, moments, and entropy of the distributions of their number of followers, posts, and so on

3)**User metadata** which include the language, time of creation, locations etc.

4)**Behavior Features** like when a user posts a tweet and time between tweets and length of tweets etc.

We are the planning to add the feature scores to Random Forests classifier which will be trained to classify the test data.

* + 1. REFERENCES

[1] B. Krishnamurthy, P. Gill, and M. Arlitt, “A Few Chirps about Twitter,” Proc. First Workshop Online Social Networks, 2008.

[2] S. Yardi, D. Romero, G. Schoenebeck, and D. Boyd, “Detecting Spam in a Twitter Network,” First Monday, vol. 15, no. 1, Jan. 2010.

[3] C. Grier, K. Thomas, V. Paxson, and M. Zhang, “@spam: The Underground on 140 Characters or Less,” Proc. 17th ACM Conf. Computer and Comm. Security, pp. 27-37, 2010.

[4] K. Thomas, C. Grier, D. Song, and V. Paxson, “Suspended Accounts in Retrospect: An Analysis of Twitter Spam,” Proc. ACM SIGCOMM Conf. Internet Measurement Conf., pp. 243-258, 2011.

[5] Emilio Ferrara, Onur Varol, Clayton Davis, Flippo Menczer, Allessandro Flammini, “The Rise of Social Bots”, Communications of ACM 59 (70, 96-104, 2016

[6] Lee, et al. Who Will Retweet This? Automatically Identifying and Engaging Strangers on Twitter to Spread Information.

[7] Erin Shellman, http://www.erinshellman.com/bot-or-not/.