Twitter Bot or Not?

Revanth Mattapalli
Department of Computer Science
Tandon School of Engineering
rm4271@nyu.edu

Varun Elango
Department of Computer Science
Tandon School of Engineering
varunelango@nyu.edu

Vignesh Ramesh
Department of Computer Science
Tandon School of Engineering
vignesh.r@nyu.edu

***Abstract*—This paper is aimed at analyzing the effectiveness and performance of different machine learning algorithms in its ability to classify twitter accounts as bots or humans. For obvious reasons, this is treated as classification problem. The paper evaluates the generalization performance and the predictive performance of all the models on future data. Based on the results we find the best suited machine learning algorithm for the given hypothesis space**

I. Introduction

Twitter is a social networking and micro blogging service, enabling registered users to read and post short messages called tweets. There are around 300 million monthly active users in posting 500 million tweets per day. Out of all the users around 23 million is estimated to be bots(automated programs which tweets, pulls data, etc without human intervention). Our aim is to build an effective prediction model that correctly classifies bots from humans.

II. Motivation

With the rising number of bots in the twitterverse, the credibility of being a popular user in the twitterverse seems to be diminishing and constant spam tweets have infiltrated our feeds. Moreover, with the advent of fake news spreading and influencing twitter users it is imperative that these bots are weeded out before they cause further harm. There are of course incredibly well written, good quality bots that posts useful information. Hence, It's imperative to find out if they are bots firsts, and then determine if they are useful or not. We, in this paper use machine learning techniques to classify if a twitter user account is a bot or not.

III. Related Work

Twitter has much popular recently and it has attracted spammers to post spam content, due to its popularity and openness. Fighting against spambot on Twitter has been investigated in recent works. Twitter users are categorized based on their tweets, entities and place. H. Kwak et al.[4] work also examined a quantitative study on Twitter by crawling the entire Twittersphere. Their work analyzed the follower-following ratio and found non-power law follower distribution and low reciprocity of tweets, which all mark a deviation from known characteristics of a human social network. Chaiji et al.[6] have shown how to maximize content propagation in ones own social network. In contrast, our approach aims at selecting a right set of bots on twitter to prevent information propagation.

Our goal is to support effective classification of the twitter users based on several features of a user. Sprout social[7] tool provides meaningful data about a twitter account such as tweet impressions,tweet activity using this data we can differentiate between bot and human. They [5] analyzed Twitter lists as a potential source for discovering latent characteristics and interests of the users. A User's feed in the Twitter consists of multiple followers and their following users' tweets. Their research indicated that words extracted from each list are representative of all the members in the list even if the words are not used by the members. It is useful for targeting users with specific interests. According to their observations, spammers send more messages than legitimate users, and are more likely to follow other spammers than legitimate users.

IV. Data

Our dataset includes, equal proportion of bot users and human users. We set up five fake twitter accounts as seeds and stated more than 90 interests for each account. This resulted in each of our accounts instantly following 300+ twitter users. Our accounts were now visible to bots that follow the same users as us. These bots immediately started following our accounts and sent us messages. Moreover, there were popular hash tags such as #fiftyshadesdarker #contentmarketing #SMM #marketing #blogging #socialmedia #growthhacking which when used also attracted few bots to start following our accounts. We manually inspected these accounts as an additional check to ensure the quality of data sample.

We also searched for famous bots mentioned in blogs and websites. Most of the bots obtained using this method were usually good, non-spam bots and were relatively easier to find. Also spam bots usually have very less followers and habitually these followers turn out to be bots. We were able to unearth a few bots using this method. Of course there were exceptions to such spam bots too, which may have a lot of followers.

A. Feature Extraction

Data collected from Twitter API are distilled in 49 features which are classified in two classes. The classes and features obtained from the twitter API is discussed next.

1) User Based Features: Features extracted from user meta data have been used to classify the users(Bot Or Human). Twitter API allows us to get the meta data of the user. Features such as followers count, friendsCount,No of tweets produced

by the user, screen name,description,location,languages know to him and settings.Explained in detail below.

- Screen Name: Provides us the twitter handle of user
- Location : Provides the location of user
- Description : A short information about the user. Some bots provide information in the description about there behavior.
- Followers Count : The number of followers this account currently has.
- Friends Count : The number of users this account is following.
- Created at : Date at which account is created by the user
- Verified: Tells us if the user is verified or not(Blue tick beside the name)
- Status count : No of tweets tweeted by user.
- Languages : The code for the users self-declared user interface language.
- Default Profile background : It is boolean value which provides the information if the user still has default profile background.
- Default Profile Picture: Also a boolean value which provides if the user has default profile picture.
- Name : The name of the user, as theyve defined it.Not necessary name of the user.
- Favorite Count:Number of tweets loved by user.
- Diversity : Length of name feature.
- Created Hour : Time at which account is created irrespective of date.
- URL : A URL provided by the user in association with their profile. Can be null.
- Account age : Number of days account is active. It is calculated from account creation date and date of last tweet created.
- Average tweets per day : As name indicates it is average number of tweets tweeted by user. It is ratio between status count and account age.
- Follower Friends Ratio : It is ratio between follower ratio and friends ratio.
- Screen name Length : Length of screen name provided by user.
- Description length : Length of desscription provided by user.
- Lexical diversity : Diversity of words used.
- Null Url : Boolean value for url
- Name ratio : Ratio between length of name and number of words in name.
- Name Bot: True if Bot is present in name.
- Description Bot : True if Bot is present in description of account.
- Number of words in name: Number of words in name of user.
- Name Length : Length of name provided by user.

2) Tweet Based Features: Features Extracted from last Tweet have been used in this class. We can get the last tweet for a particular user from Twitter API.From last tweet(Status) we can get features such as text of tweet, time of tweet tweeted, re-tweet or not, if tweet is in reply to some other user,Favorite count,Hash tags included in text,links included in text.More features are explained below.

- Truncated : If the tweet text is cut short to accommodate only 140 characters(Tweet length)
- Text : Provides tweet text tweeted by user.
- In reply to tweet id : Tweet id of main tweet
- Id : A unique id is given to each tweet.
- Favorite Count:No of users have loved this particular tweet.
- Coordinates : Provides the location from where the tweet is tweeted.
- User Mentions : Twitter handle of users mentioned in tweet.
- Hashtags : Hashtags present in the tweet.
- Retweet count : Number of times tweet is retweeted.
- Created at : Time of tweet tweeted
- Retweeted :Provides the information if the tweet is retweeted by some other user.
- In reply : True, if tweet ext is in reply to different user

In addition to above features mentioned, we collected last 100 tweets of each user and corresponding parameters. Features extracted from the data is explained further in below section.All feature value mentioned below are for last 100 tweets.

- In-reply count: No of tweets tweeted in reply to some other user in last 100 tweets.
- Retweet count: Number of tweet which are retweet.
- Favourite count: Number of tweets liked by different user.
- User mentions: No of users mentioned in tweets.
- Created dates: Creation dates for last 100 tweets
- Texts: List containing last 100 tweet text.
- Average in-reply count:Ratio between in-reply count and 100.
- Average retweet count: Ratio between retween count and number of tweets
- Average favourite count: Ratio between favourite count and number of tweets.
- Average user mentions: Ratio between favourite count and number of tweets.
- Tweet Days: Dictionary which gives distribution of last 100 tweets with respect to day of week(Monday,Tuesday,etc)
- Tweet hours:Dictionary which gives distribution of last 100 tweets with repect to hour of day.
- Tweet Days ktest: Performed Kolmogorov-Smirnov on tweet days distribution.
- Tweet hours ktest: Kolmogorov-Smirnov on hours days distribution.

Figure 1 describes entropy of training and test data. No of humans present in the data is almost equal to that of bots. So we have high entropy.

Figure 3&4 are frequency graphs for number of followers for bot and human respectively.It can be seen that followers

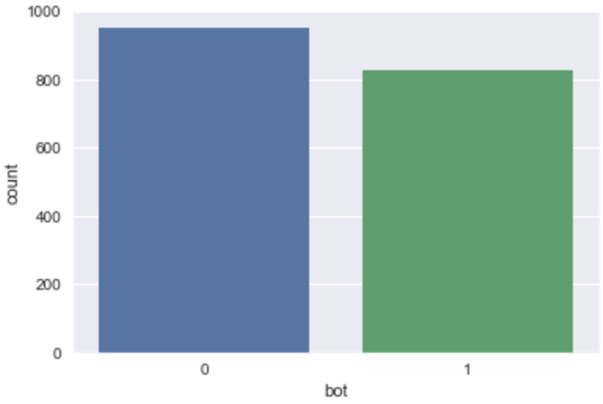

Fig. 1. No of Humans vs Bots

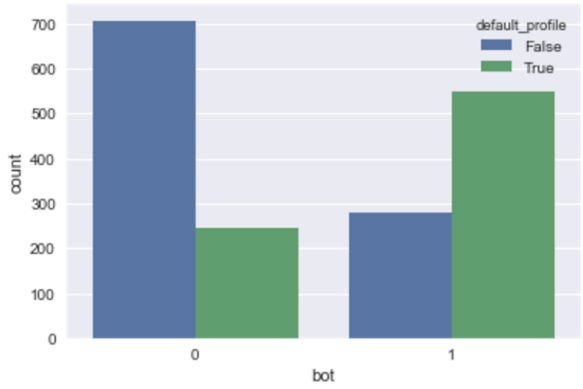

Fig. 2. No.of Default Profile on Bot and Humans

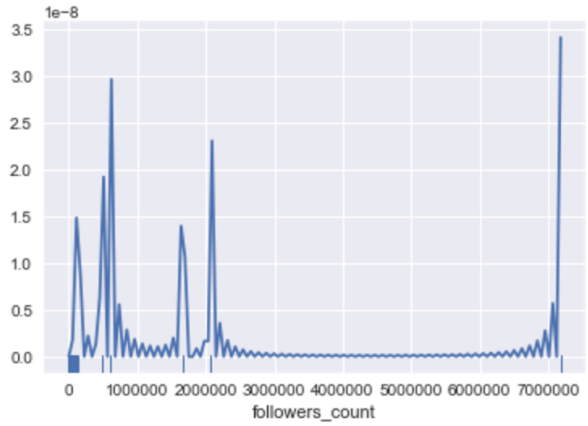

Fig. 3. No.of Followers Count on Bot

Figure 2 describes relation between user profile and default profile picture. It can be noted from the graph that most bots have not changed default profile picture. It can help in determining profile(human or bot) of the user.

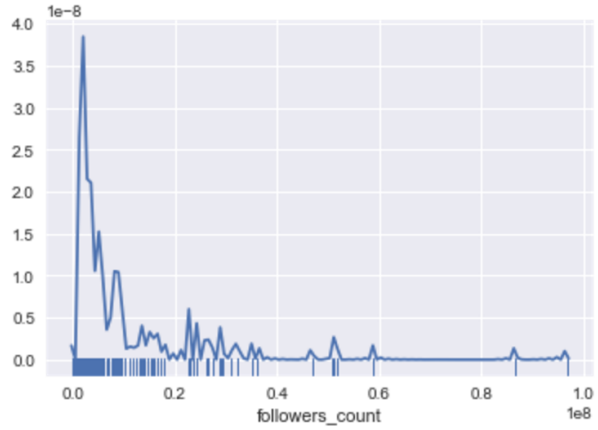

Fig. 4. No.of Followers Count on Human

Fig. 5. Verified vs friends Count

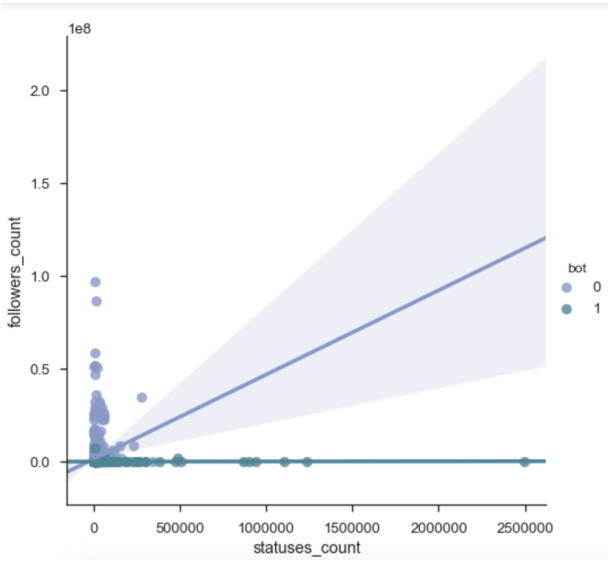

Fig. 6. Trends of bot on status Count and followers Count

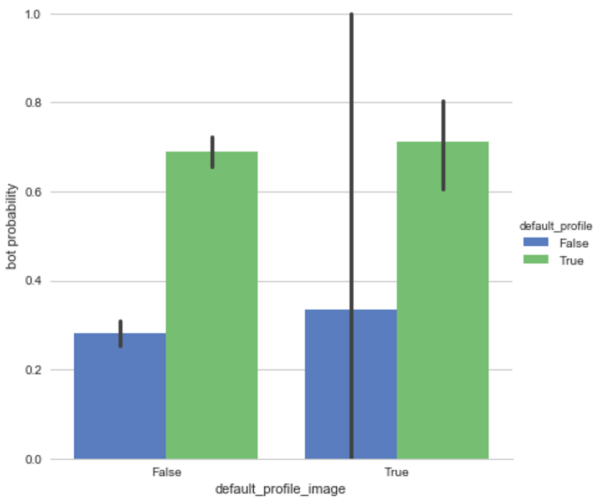

Fig. 7. Default Profile vs Default Profile Image

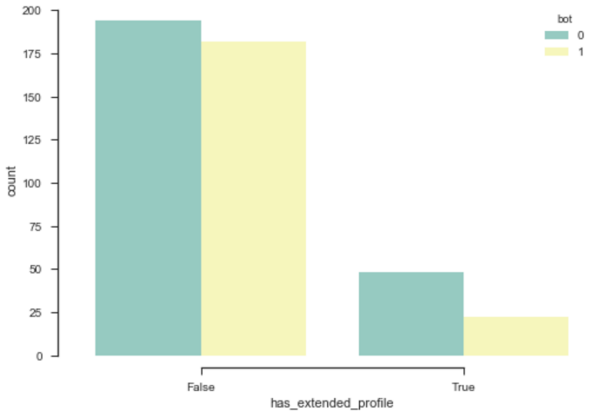

Fig. 8. No.of HasExtendProfile on bots and Humans

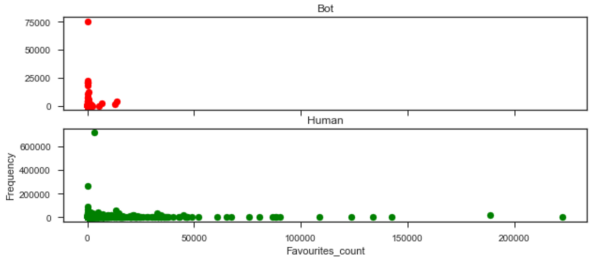

Fig. 9. Frequency on Favorites count

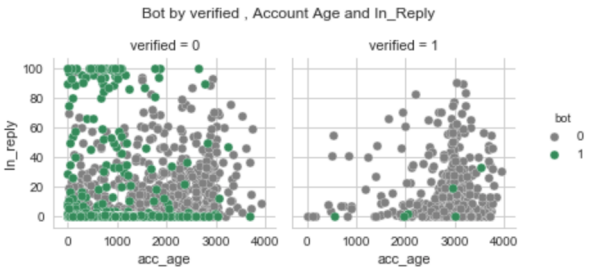

Fig. 10. Bot by verified,account age and no of in-reply tweets

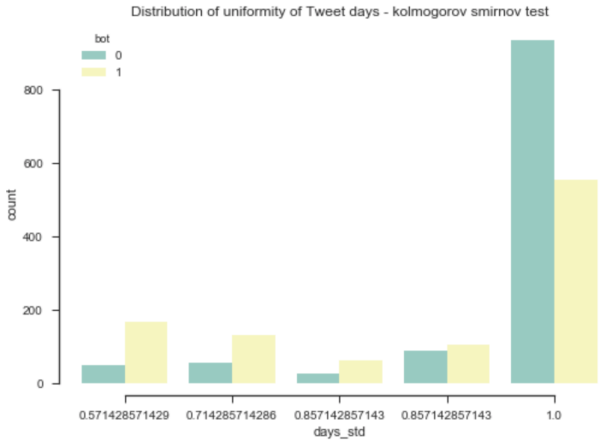

Fig. 11. Distribution of tweet days

count for human is generally high. Followers count and probability of account being human is positively correlated. Where as for bot followers count is generally low. It can be noted that probability of account being bot and followers count is negatively correlated.

Figure 5 depicts relationship between number of users this account is following and account being verified. It can be seen that even though account has high friends count,account is human if it is verified.Which in general is behavior of bots

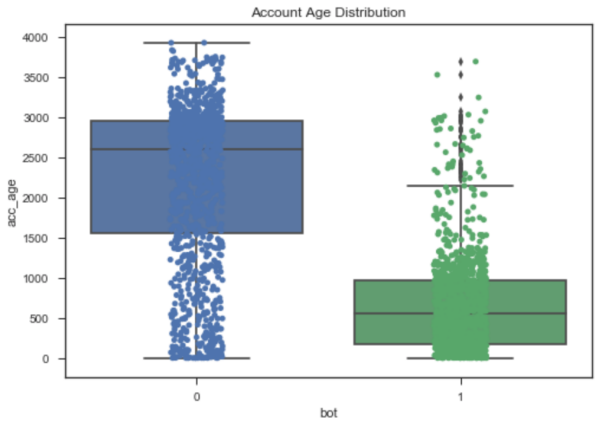

Fig. 12. Box graph for Account age

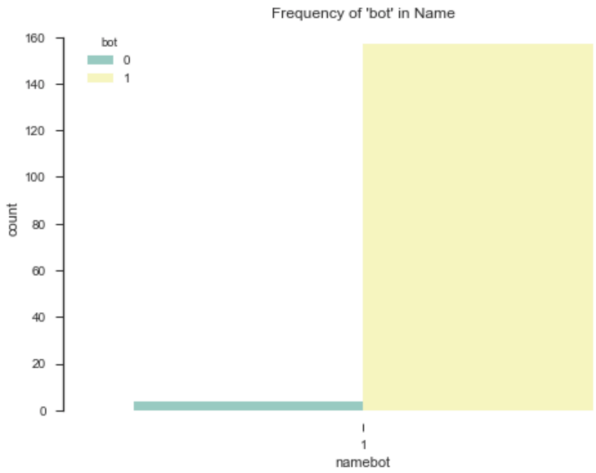

Fig. 13. Distribution for account name containing bot

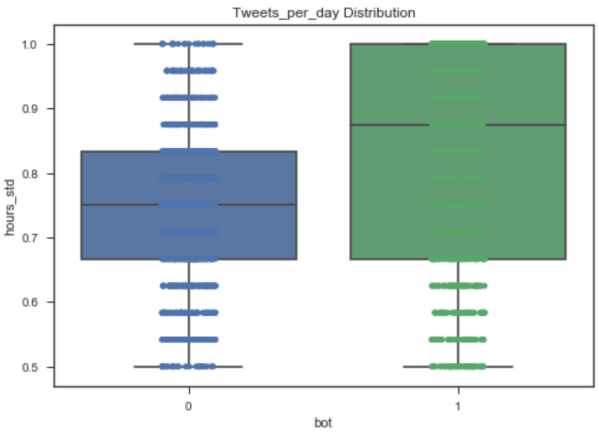

Fig. 14. Distribution for tweets tweeted per day

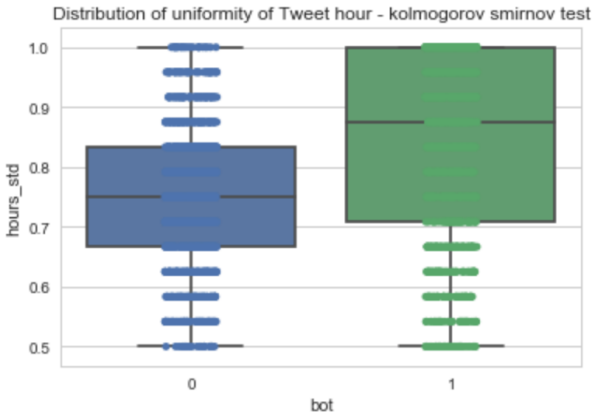

Fig. 15. Distribution of tweet hours

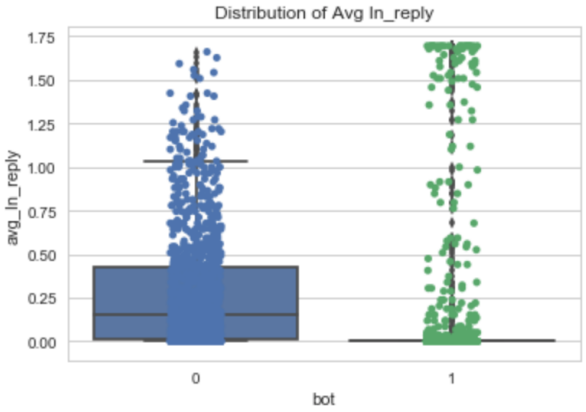

Fig. 16. Distribution of average In-reply count

Figure 6 depicts the relationship between no of tweets tweeted by the user and followers count. It shows that if No of tweets for particular user is high and follower count is low then there are high chances that particular account is bot.

Figure 7 describes relationship between default picture and default background. It can be seen that probability of the user been bot is high if the user has default profile picture and default profile background. In general that probability of the profile being bot is high if the account has default profile picture can be verified from graph.

Figure 8 describes influence of account containing extended profile.It can be seen from graph that bot account have less probability of having extended profile.

Figure 9 depicts account having low favorite count are more probable to be bot. So favorite count and probability of account being human is positively correlated.

Figure 10 describes relationship between verified,account age and in-reply count(data collected from twitter API). It can be seen that if account age is low and in-reply data is high then account is not human.Where as if account age is high and in-reply count is not low(medium to high) then the account is more probable to be human.

Figure 11 describes of uniformity of tweets tweeted on days of week. It can be seen from the graph that if uniformity is 1 then account is more probably bot.

Figure 12 depicts relationship between account age and account behavior. It can be seen, behavior of account with low age is more probable to be bot. From graph it is implied that 75 percent of bot data has low account age. And remaining 25 percent data has account age less than mean of human account age.

Figure 13 describes relationship between name bot and bot account. It can be seen from graph that if account have 'bot' keyword in name then account is bot.

Figure 14 is distribution between tweets per day and account behavior. It can be inferred that if tweets per day is high then behavior of account is not human. It is understood from graph that 50 percent of bot accounts have comparatively higher tweets per day.

Figure 15 is distribution of hour tweet tweeted and account behavior.IT can be seen that more uniform data is more probable not be human. it can be seen that higher number of bots have uniform tweet hours.

Figure 16 shows distribution of average in-reply counts for twitter accounts. It can be seen that majority of bot accounts don't have in-reply tweets. Only few out-liner account bot account contain in reply data.

V. Algorithm(s) Used

The available data was split into 80% training set and 20% testing set. Discrete value attributes were encoded with numrical values using out of the box LabelEncoders. In order to identify important features that influence prediction we used recursive feature elemination with cross validation(fig 17). Our base estimator for rfecv was random forest. We also plotted the auc curve for each feature to determine its importance adn filtered top 15 features as shown in figure 18. We finalized the set of features by training logitic regression and and decision tree with both combintation of feature sets to determine the best feature peroforming feature set. The results are shown in fig 19 sets These optimal feature set of top 20 features is expected to reduce overfitting and increase accuracy. The following are the features selected friends count,followers count, favourite scount,verified,default profile,listedcount, ff ratio,tweets per day,acc age,statuses count,nameratio,name bot bool,desc bot bool,n count,n len,In reply,retweet count,fav count,total usrmention.

For baseline, majority class classifier was used which had an accuracy of 0.5037.

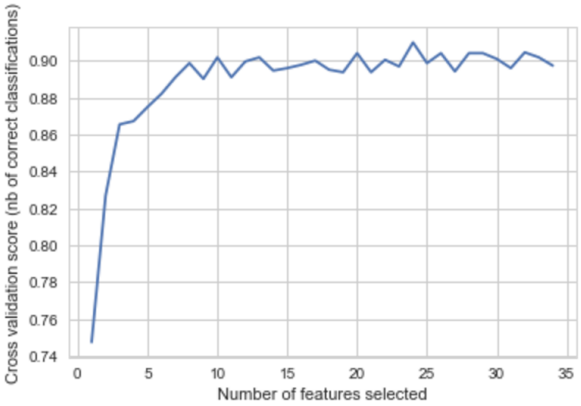

Fig. 17. Feature Selection

A. Decision Tree

A Decision tree was built using the features mentioned and had the overall prediction accuracy for the test data was .85 and the cross validation had a mean of .82. The confusion matrix had a precision of 0.840 for both classes and a recall of 0.867 for both classes. The false negatives were 31 out of 461 total test records. Similarly there were 37 false positives. 10 Bot accounts that were misclassified as humans were verified

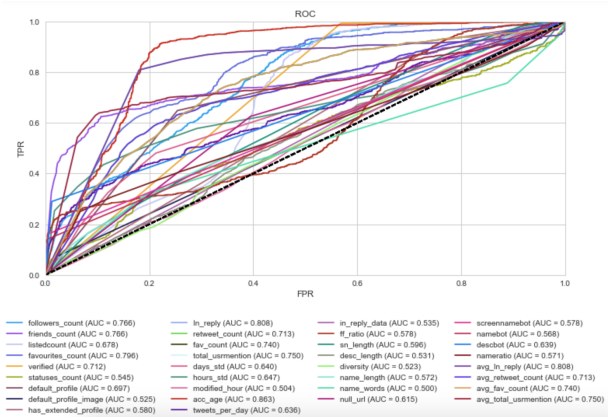

Fig. 18. ROC

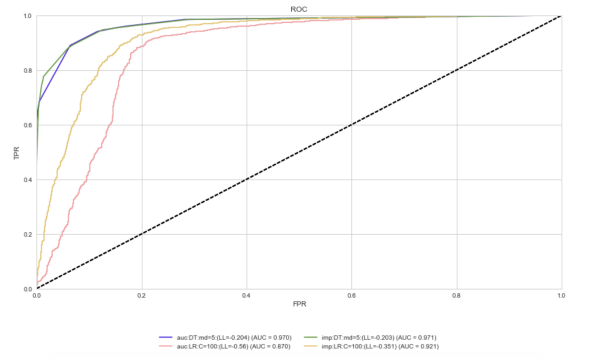

Fig. 19. ROC for Decision Tree and Logistic Regression

bots. Remaining bot accounts had either follower counts or friends count or favorites counts or listed count similar to the mean of the respective features in human accounts. To reduce the influence of these features pruning techniques need to be used. As identifying the optimal tree in post pruning techniques is a Np complete problem we used only pre-pruning techniques by varying the max-depth of the tree and impurity threshold . Optimal depth of the tree was found to be 6 and impurity threshold 1e-8. The decision tree was now able to predict with an improved accuracy of 0.8611 and the false negatives reduced to 23 records.

The Grid Search in Scikit-learn to chose the parameters for the decision tree. The best hyperparamters predicted by the Grid search model also has the same parameters that we had used before.

- criterion: gini
- max depth: 6
- max features: 6
- min impurity split: 1e-08
- min samples leaf: 8

We built the decision tree with these paramaters and the tree produced is shown in the Fig. 11. With the friends count as the root of the tree.

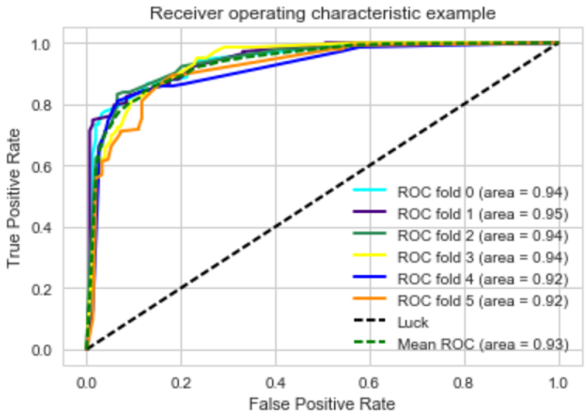

Fig. 20. Receiver Operating Characteristic

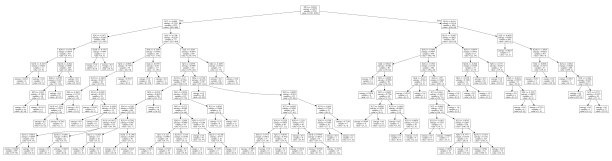

Fig. 21. Decision Tree

B. Logistic Regression

As this classifier assigns weights to features, we needed to scale the data inorder to reduce the influence of high scale features skewing prediction results. We used the standardscalar function from the sklearn package. The tolerance value of the model was set to 1e-4 and stochastic gradient descent was used to find optimal weights. Also inorder to reduce overfitting, l2 regularization was used to assign penalities to weights and the resulting accuracy was around 0.82. Upon using Limited-memory BFGS optimization method to find optimal weights the accuracy improved to 0.88. l1 regularization had a similar impact on the prediction as l2. We chose l2 as l1 has a possibility of getting multiple solutions and more stable when compared to l2.

On Further analyzing the data it was found that several polynomial derived data improved prediction accuracy. The out of the box polynomial feature pipline with a polynomial degree of 2 significantly improved the prediction accuracy to 0.89.

Similar to the grid search done for decision trees, a grid search for logisitc regression was done and it obtained optimal hyperparameter settings and had a prediction accuracy of 0.89 and across validation mean of 0.88

C. Multilayer perceptron

This supervisd neural netowrk algorithm uses a tanh activation function for each neuron as it provides stronger gradients.

$$f(z) = \frac{e^z - e^- z}{e^z + e^- z}$$

MLP is sensitive to different scales of data, similar to that of logistic regression, hence we scaled the features using standardscalar. The optimal number of hidden layers used was 2, with first layer having around 16 neurons and the second layer having around 10 neurons. The learning rate was to 1e-3 to and the error tolerance level for convergence was set to 1e-7. The weight optimization algortihm that was used is lbfgs. MLP with these hyperparameter settings resulted in a prediction accuracy of 0.89 and a cross validation mean of .87. Altough it can learn non-linear function approximators, as there weren't enough data and features avaiable, MLP wasn't able to push the accuracy beyond .89

D. Ensemble methods

Among Several contemporary ensmeble techniques randomn forest classifier was our go to algorithm. When the random forest classfier was trained with a max depth of 3 and the impurity criteria set to entropy, we obtained an accuracy of 0.89 and a cross validation mean of .88. when the impurity criteria was set to gini and the max depth was set to 6 a prediction accuracy of .91 and a cross validation accuracy of .90 was obtained.

Inorder to further improve the accuracy we decided to boost randomn forest by using another ensemble technique on top of it with ada boosting. The number of estimators used was set to 10 and the algorithm set to SAMME.R. The overall prediction accuracy obtained was 0.92 and a cross validation mean of .91

To make our model further sensitive to noisy data and outliers we used voting classfier with logisitc regression with gradient boost and random forest with ada boost as our base estimators. It was found out that account age might might be factor that overfits the data resulting 35 false postives. Hence the base estimators were trained with the two different feature sets one with account age and one wihout account age. These 4 estimators were used as an input to the voting classfier. The overall prediction accuracy was .93

VI. Result

The evaluation is based on the following metrics:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F - Measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

where TP, FP and FN are the numbers of true positive, false positive and false negative results respectively. We used cross validation mean and the lowest accuracy score from the cross validation score to evaluate our models. Below table showes the crooss validation mean and f1 scoores for each model that we had tried.

The overall accuracy was higest for our random forest with ada boosting and voting classifier. These models performed well compared to others as they were more sensitive to noisy data and outliers.

'

TABLE I
CROSS VALIDATION MEAN AND F1 SCORE

Algorithm used	CV mean	F1 score
MLP Neural network	0.88	0.86
Logistic Regression(L2)	0.88	0.89
Logistic Regression(L1)	0.89	0.90
Random Forest with Ada Boosting	0.92	0.92
Decision Tree	0.89	0.89
Voting classifier	0.93	0.93

VII. CODE

We have included all the codes for Data fetching, Preprocessing and ML Classification models in the following git-hub link: https://github.com/Vignesh6v/Twitter-BotorNot/

VIII. VIDEO LINK

Presentation video have been hosted on Youtube on the following link https://www.youtube.com/watch?v=4wtghRGhNVQ

IX. EVALUATION

Based on the results obtained using the metrics specified in result section and accuracy result of test data obtained from kaggle, we think that Random forest algorithm using ADA boosting have performed better than remaining algorithms we have tried. While using decision tree algorithm for predicting behavior of account, we were struck with an accuracy of 89 percent for test data and 84 percent of accuracy for kaggle test data. Then we started working on logistic regression combining it with grid search for selection of best features,accuracy was around 90 percent. So we started using ensemble techniques such as random forest for better efficiency. Upon creating model we further enhanced the accuracy using ADA boosting on random forest estimator where the accuracy score of the model was 95 percent and accuracy score on kaggle data was 96 percent.

X. CONCLUSION

In this paper we have studied behavior of account using features provided by twitter API and further modification of data.To better understand the behavior of twitter account we have used various machine learning techniques using different preprocessing techniques.As different types of estimators and refining of data we come closer for comprehensively understanding nature of account. In order to increase efficiency of model we can further study features of followers account and there connections,semantic analysis of tweets tweeted,setting up honey pot.

ACKNOWLEDGMENT

We would like to express our gratitude to Prof. Gustavo Sandoval for his advice and guidance throughout the project.

REFERENCES

[1] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer and Alessandro Flammini, *The Rise of Social Bots.* X, X, Article XX (201X), 11 pages.

[2] Kyumin Lee, Jalal Mahmud, Jilin Chen, Michelle Zhou and Jeffrey Nichols, *Who Will Retweet This? Automatically Identifying and Engaging Strangers on Twitter to Spread Information*. Harlow, England: Addison-Wesley, 1999.

[3] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

[4] H. Kwak, C. Lee, H. Park, and S. Moon, *What Is Twitter, a Social Network or a News Media?* Proc. 19th Intl Conf. World Wide Web, pp. 591-600, 2010

[5] I.-C.M. Dongwoo Kim, Y. Jo, and A. Oh, *Analysis of Twitter Lists as a Potential Source for Discovering Latent Characteristics of Users,* Proc. CHI Workshop Microblogging: What and How Can We Learn From It?, 2010.

[6] Chaoji, V., Ranu, S., Rastogi, R., and Bhatt, R. *Recommendations to boost content spread in social networks.*, In WWW, 2012.

[7] Human-Bot Interactions: Detection, Estimation, and Characterization , Onur Varol, Emilio Ferrara, Clayton A. Davis, Filippo Menczer, Alessandro Flammini

[8] *http://sproutsocial.com/insights/twitter-data*