International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST

Twitter Data Analysis for BOT Classification

Shweta Patil, Aparna V. Mote
Department of Computer Engineering, Zeal College of Engineering and Research,Narhe, Pune-411041, Maharashtra, India

Article Info
Volume9, Issue 3
Page Number: 586-588
Publication Issue
May-June-2022
Article History
Accepted :01May2022
Published :12 May 2022

ABSTRACT
With millions of users, Twitter is one of the most well-known microblogging platforms. Users are free to write about anything they like, including politics, sports, culinary, fashion, etc. Various assaults, including the dissemination of disinformation, phishing links, and malware, have targeted Twitter. Tweets must be posted by actual people and not by Twitter bots. The existing approaches leverage the user's tweets to make this determination, placing more emphasis on accuracy than efficiency. In this study, a feature engineering pipeline has been created to effectively distinguish between Twitter bots and actual users using user metadata such as default name, description, etc. There has been discussion of several machine learning technique algorithms.An accuracy of 98% using the proposed approach was obtained. The performances of various classifiers like the Decision tree classifier, Random Forest classifier, Multinomial Bayes classifier, KNN, and Logistic Regression classifier are compared to find the best classifier.

Keywords—Twitter, Twitter Bots, Metadata, Feature engineering, Machine Learning, Classification.

I. INTRODUCTION

Users submit and engage with messages known as "tweets" on Twitter, a microblogging and social networking site [1]. As of January 2022, there were 76.9 million users of the microblogging site. India was third with about 23.6 million users [2]. One of the most well-known social networks in the world, Twitter continues to be a popular marketing platform. Quick-fire tweets may spark real conversations anywhere in the globe on Twitter, which has always been a social medium. Twitter, which has millions of users, is the finest platform for interacting with clients. Writing, speaking, and providing comments are all combined on the platform known as Twitter. A community or the entire country may be made aware of all the data and information, in addition to friends and family.It might help someone start their own self-published journal or give advice and assistance to others. Twitter, on the other hand, is the perfect platform for any government agency or company stream to broadcast information because of its enormous readership [3].

Unregistered users can only view tweets that are publicly available. Registered users can write, like, and retweet tweets. Users can tweet via the Twitter website, third-party apps (such as smartphones), or SMS (which is available in some areas). Users can "follow" other users to receive their tweets, and subscribers are referred to as "followers." Other users can rebroadcast individual tweets to their own streams [1].

A Twitter bot is a computer software that tweets automatically. They're set up to tweet, retweet, and follow other people's accounts. According to a recent research, there were 20 million bogus accounts on Twitter. Botnets frequently use Twitter bots. A botnet is a large group of automated accounts that collaborate to make themselves look authentic by like and following each other as if they were real. Twitterbots can sway public opinion on culture, goods, and political agendas by automating the production of large numbers of tweets that mimic human conversation. The societal ramifications of these Twitterbots on human perception are significant. Twitter bots are frequently used by cybercriminals to concurrently disseminate dangerous material, such as malware, to huge groups of Twitter users [1].

This paper discusses several machine learning algorithms for identifying Twitter spam bots. The CrowdFlower authors' annotated MIB dataset [4] of real and spammy Twitter accounts has been used. This dataset includes tweets from real accounts as well as bogus followers and social spambots. On features like statuses count, follows count, friends count, favourites count, screen name, location, and verified profile, much feature engineering has been done. The decision tree classifier, Random Forest, Multinomial Bayes, KNN, and Logistic Regression are some of the classification methods that have been applied. Each classifier's accuracy is calculated, and the method with the highest accuracy is chosen.

II. LITERATURE SURVEY

There exist several previous surveys related to Twitter bot detection. However, each one has its limitations and strengths.

In their research [5], Ranjana Battur and Nagaratna Yaligar take a dataset from Kaggle and extract characteristics based on the Spearman correlation coefficient. It includes information on the user's friends, followers, location, screen name (used for online communication), verified status (if the user has been verified), favourite status (used for liked tweets), url, id, description, and listed count, among other things. The implemented algorithms are Decision Tree, which provides accuracy of 87.85 percent, Multinomial Nave Bayes, which provides accuracy of 69.76 percent, Random Forest, which provides accuracy of 86.19 percent, and Bag of Words, which provides accuracy of 95.24 percent.

The paper "Tweet-Based Bot Identification Using Big Data Analytics" [6] by A. Derhab, R. Alawwad, K. Dehwah, N. Tariq, F. A. Khan, and J. Al-Muhtadi offers a taxonomy that categorises the state-of-the-art machine learning algorithms for tweet-based bot detection. In order to combat tweet-based botnets and reliably discriminate between human accounts and tweet-based bot accounts, this research has concentrated on large data analytics, particularly shallow and deep learning. In addition, tweet-based bot detection methods using shallow and deep learning are discussed, along with their effectiveness. With several datasets, the accuracy of each shallow and deep learning approach has been illustrated.

In their study [7], Kabakus, Abdullah Talha, and Kara offered a brief comparative overview of the research on Twitter spam detection conducted between the years of 2009 and 2015. Within four categories—account-based, tweet-based, graph-based, and hybrid-based—they discussed several detection techniques.

The account-based approaches were demonstrated to take use of user profile metadata, including followers and following count, as well as other derived attributes, including account age. While it has been demonstrated that parameters like the distance and degree of connectedness between individuals may be employed for spam identification in graph-based approaches. The survey, however, largely concentrated on identifying spam utilising URL and its derived properties, such as length and domain name, in tweet-based approaches. To detect a spam user, posted URLs were analyzed and classified as malicious or benign. Besides this, the authors highlighted overlooked features that were argued to improve spam detection.

Nivranshu Pasricha, Conor Hayes in their paper [8] make use of Digital DNA Compression to detect Twitter bots. The authors offer a method for detecting bot-like behaviour among Twitter accounts by examining their previous tweeting behaviour. They based on an existing Twitter account analysis technology called Digital DNA. Digital DNA simulates the behaviour of Twitter accounts by storing a user's post history as a sequence of characters similar to a DNA sequence. A lossless compression method on these Digital DNA sequences is used in this approach, and the compression statistics are used as a measure of predictability in the behaviour of a collection of Twitter accounts. They used the compression statistics to create a simple two-dimensional scatter plot to graphically display the posting behaviour and categorise the data.

Loukas Ilias, Ioanna Roussaki [9]introduce two methods targeting this that are mainly based on Natural Language Processing (NLP) to distinguish legitimate users from bots. A feature extraction methodology is provided in the first method for detecting accounts that send automated messages. The subset of characteristics chosen is given into machine learning algorithms after using feature selection techniques and coping with skewed datasets. A deep learning architecture is provided in the second technique to determine if tweets were posted by actual people or created by bots. The proposed methods were tested in a series of tests using two large actual Twitter datasets, and they show significant improvements over other current strategies for detecting fraudulent individuals on social media.

M. Fazil and M. Abulaish [10] applied Random Forests, Decision Trees, and Bayesian Networks as classification models for bots' detection on Twitter. Furthermore, they employed a set of features, six of which have not been used before, that can be grouped into four main categories: metadata, content, interaction & community. For dealing with the imbalanced dataset, they applied SMOTE, which constitutes an oversampling technique.

A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu and C. Yang [11] presentanother approach that also introduces new features that had not been used before is proposed. They identified 15 new features and employed four machine learning classifiers for detecting spam tweets. As mentioned by the authors, these features exploit the behavioral entropy, profile characteristics, bait analysis, and the community property observed for modern spammers.

Jison M Johnson and the co-authors [12] have implemented a web application to detect Twitter bots. Machine learning algorithms are used to detect Twitter bots. They look at things like tweets, likes, and retweets, among other things. The data is then utilised to train their model utilising Decision Trees and Random Forest machine learning approaches. They utilised the flask server to connect their model to the web content. With reasonable accuracy, their framework identifies whether the user belongs to a human account or a bot.

Feng Wei, Uyen Trang Nguyenuse recurrent neural networks (RNN), specifically bidirectional Long Short-term Memory (BiLSTM), to efficiently capture features across tweets, in their paper [13]. Their work develops a recurrent neural model with word embeddings to distinguish Twitter bots from human accounts, that requires no prior knowledge or assumption about users' profiles, friendship networks, or historical behavior on the target account.

N. Narayan[14]have used three machine learning algorithms to detect whether the account is fake or real, which are Decision Tree, Random Forest, and Multinomial Naive.The classification performance of the algorithms is compared with their accuracy. The accuracy given by the Decision tree algorithm is 93%, the Random Forest algorithm is 90% and the Multinomial Naive Bayes is 89%.

III. PROPOSED SYSTEM

Fig.1 shows the block diagram of the proposed System. Preprocessing Techniques like feature extraction and feature engineering are applied on the Twitter dataset. Classification algorithms are further implemented on the preprocessed data.

A. Dataset

The dataset for real and spam Twitter accounts from the MIB [4] has been utilised. This dataset contains information on 11017 people, including real accounts, classic spambots, social spambots, and phoney followers. This dataset has 42 characteristics. Table 1 contains a summary of the dataset.Some of the features present in this dataset are - statuses count, followers count, friends count, favorites count, screen name, location, verified profile, etc.

Not all the features in the dataset are useful and some of them are not directly useful either, hence the relevant features need to be extracted and some others need to be engineered.

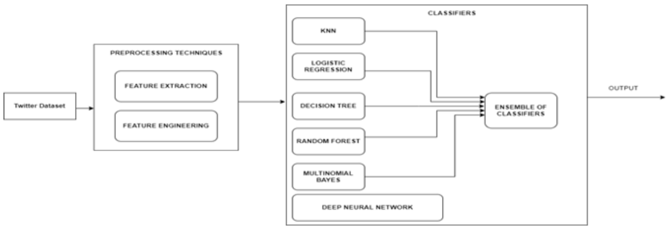

Fig. 1Proposed System

B. Preprocessing

Data preparation is the process of putting raw data into a format that is comprehensible. In the suggested method, we pre-process the data to fill in blank entries and remove ones that are inaccurate. Additionally, normalisation is carried out to improve accuracy. As part of preprocessing, we also use feature engineering and feature extraction.

TABLE I SUMMARY OF DATASET

group name	description	accounts	tweets	year
genuine accounts	verified accounts that are human-operated	3,474	8,377,522	2011
social spambots #1	retweeters of an Italian political candidate	991	1,610,176	2012
social spambots #2	spammers of paid apps for mobile devices	3,457	428,542	2014
social spambots	spammers of products on	464	1,418,626	20

#3	sale at Amazon.com			1 1
traditional spambots #1	training set of spammers used by C. Yang, R. Harkreader, and G. Gu.	1,000	145,094	2 0 0 9
traditional spambots #2	spammers of scam URLs	100	74,957	2 0 1 4
traditional spambots #3	automated accounts spamming job offers	433	5,794,931	2 0 1 3
traditional spambots #4	another group of automated accounts spamming job offers	1,128	133,311	2 0 0 9
fake followers	simple accounts that inflate the number of followers of another account	3,351	196,027	2 0 1 2

C. Feature extraction and engineering:

As discussed previously, all the features from the datasetcannot directly be used, because they are either not relevant or they need to be transformed to some better usable form. This is why feature extraction and engineering is being performed.

To start with the authors of this paper perform the Spearman Correlation on the given data and select the features that have a high correlation with the dependent variable. The correlation values of these features are given in Table 2.

TABLE II FEATURE CORRELATION

Features	**Bot**
screen_name_binary	-0.033576
location_binary	0.321255
desc_binary	0.432417
def_profile_binary	0.324657
def_profile_img_binary	-0.023526
statuses_count	-0.656845
followers_count	-0.404984
friends_count	-0.281992
favourites_count	-0.904498

1) Spearman Correlation: A non-parametric test called Spearman rank correlation is used to determine the degree of relationship between two variables. When the variables are measured on a scale that is at least ordinal, the Spearman rank correlation test is the suitable correlation analysis [15]. It makes no assumptions about the data distribution.

As the selected features are fewer, the next step is to engineer features from already existing ones. Features are engineered for those features that have the text and hence cannot be directly processed by the ML algorithms. The following features were engineered - Table 3.

TABLE III SUMMARY OF SELECTED FEATURES

Features	Data Type	Description
ID	Int64	The full ID is composed of a timestamp, a worker number, and a sequence number.
Screen Name	object	This contains display name which is a personal identifier on Twitter.
Statuses Count	int64	Specifies the number of Tweets to try and retrieve, up to a maximum of 200 per distinct request.
Followers Count	int64	This specifies the number of followers per account has.
Friends Count	int64	This specifies the number of Friends per account has.
Favourites Count	int64	The favorite_count provides the number of times the tweet has been favourited
location	object	This specifies location of the users.
default_profil e	float64	This specifies the profile of the user.
Description	object	This contains description of the Tweet.

After engineering the above features,Spearman Correlationis used to find the relevance of the engineered features for predicting the dependent variable. The results of the same are mentioned in the table (same as the one in feature extraction)

The aforementioned features are based on the metadata of the user's profile; however, certain features are also being developed depending on how the user tweets. The total number of tweets and the time delta feature (the number of days between the first and latest tweet) are these features. These characteristics are crucial because some clever bots have the necessary profile information to outwit actual users, but there is usually a pattern to their tweet behaviour, such as when they tweet excessively frequently, when they are very periodic in nature, when they reply to tweets with particular keywords, etc.

D. Classification

After feature extraction and feature engineering classification are implemented. Well-known ML classification techniques are used-Decision Tree classifier, Multinomial Naïve Bayes classifier, Logistic Regression, Random Forest classifier, and KNN. An ensemble of classifiers and a Neural Network is also implemented.

1) Decision Tree Classifier: A selection For categorization tasks, Decision Tree is a supervised learning technique. Internal nodes represent dataset attributes, branches represent decision rules, and each leaf node represents the outcome in a tree-structured classifier. The Decision Node and the Leaf Node are the two nodes of a Decision tree. The test or decision is based on the properties of the given dataset [16]. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features [17].

The Decision Tree Classifier requires two arrays as input: an array X that contains the selected features, and an array Y that has the bot value (either 0/1 for non-bot and bot, respectively).

The model can then be used to predict the training and testing data after it has been fitted. The accuracy of training and testing is then calculated.

2) Multinomial Naive Bayes: The Multinomial Naive Bayes algorithm is a probabilistic learning approach popular in Natural Language Processing (NLP). The algorithm guesses the tag of a text, such as an email or a newspaper story, using the Bayes theorem. It calculates each tag's likelihood for a given sample and outputs the tag with the highest probability. A naive Bayes classifier is a collection of several methods that all follow the same principle: each feature being classified is unrelated to any other feature. A feature's presence or absence has no bearing on the presence or absence of another feature [18].

The Bayes theorem determines the probability $P(c|x)$, where c is the class of probable outcomes and x is the supplied case to be identified, which represents some specific characteristics[19] . $P(c|x) = P(x|c) * P(c) / P(x)$.

3) Logistic Regression: The supervised learning classification method logistic regression is used to predict the likelihood of a target variable.

In basic terms, the dependent variable is binary in nature, with data represented as 1 (representing success/yes) or 0 (representing failure/no).Mathematically, a logistic regression model predicts $P(Y=1)$ as a function of X [20].

When a decision criterion is included, logistic regression transforms into a classification procedure. The threshold value is a crucial feature of Logistic regression, and it is determined by the classification issue itself [21].

4) Random Forest Classifier: Random Forest is a well-known supervised learning technique-based machine learning algorithm. It is used to address categorization difficulties in machine learning. It is based on the notion of an ensemble.

According to the term, Random Forest is a classifier that incorporates several decision trees on distinct subsets of a given dataset and averages them to raise the projected accuracy of that dataset.The random forest collects forecasts from each tree and predicts the ultimate output based on the majority votes of projections, rather than depending on a single decision tree. The more trees in the forest, the higher the accuracy and the lower the risk of overfitting [22].

Random forest classifiers use a decision tree as their principal component. The decision tree is a hierarchical structure built from the characteristics of a data collection. A measure associated with a subset of the characteristics is used to partition the decision tree into nodes [23].

5) K-Nearest Neighbor (KNN): One of the simplest Machine Learning algorithms is K-Nearest Neighbor, which is based on the Supervised Learning methodology. The K-NN method places the new case in the category that is most similar to the existing categories on the assumption that the new case/data and previous cases are comparable. The K-NN approach keeps track of all the data that is available and categorises additional data points depending on how closely they resemble the existing cheval. This means that the K-NN approach can swiftly classify fresh data into the appropriate category. The K-NN approach may be used for classification jobs [24].

6) Neural Network (keras model): Keras is an open-source API that may be used to solve a range of machine learning and deep learning challenges. A typical Keras model is made up of several training and inferential layers. Keras' sequential model was utilised. Each network layer takes just one input and transmits only one output. In this situation, the activation function is 'ReLu,' which stands for rectified linear activation unit.

7) Adaboost Ensemble classifier: AdaBoost is a method of ensemble learning was developed to improve the efficiency of binary classifiers. AdaBoost employs an iterative strategy to improve poor classifiers by learning from their mistakes.

Ensemble learning is a method of combining numerous basic algorithms to create a single optimal prediction algorithm. AdaBoost (Adaptive Boosting) is a prominent boosting approach that seeks to construct a strong classifier by merging many weak classifiers. A single classifier may not be able to reliably forecast the class of an item, but a powerful model may be formed when numerous weak classifiers are put together and each one learns from the others' incorrectly categorised objects [25].

8)Voting Classifier (Ensemble): A Voting Classifier is a machine learning model that learns from a group of models and predicts an output (class) based on the class having the best chance of becoming the output.

It simply sums up the results of each classifier fed into the Voting Classifier and predicts the output class with the most votes. Rather of building and testing separate specialised models, the idea is to establish a single model that trains many models and predicts output based on the aggregate majority of votes for each output class [26].

IV. RESULTS

Here the Training and Testing accuracy are presented on the given Twitter dataset against all the mentioned classifiers, Neural Network, and Ensemble of classifiers-Table 4.

Fig.2 shows the ROC curves and their ROC values for each classifier respectively.

TABLE IV RESULTS

Classifiers	**Training Accuracy**	**Test Accuracy**
Decision Tree Classifier	97.718%	98.034%
MultinomialNB	87.044%	87.992%
Random Forest Classifier	97.29%	97.48%
KNeighborsClassifier	96.32%	95.10%
Logistic Regression	90.37%	91.10%
Neural Networks	95.28%	95.78%
Ensemble(AdaBoostClassifier)	97.34%	98.17%
Ensemble(VotingClassifier)	96.73%	97.12%

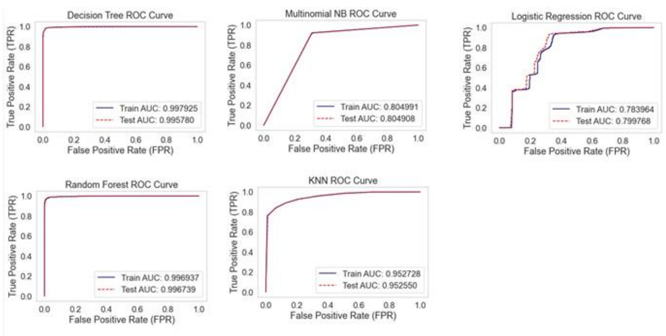

Fig. 2 ROC plots for the prediction on the entire dataset

V. CONCLUSION

One of the most widely used social networking sites is Twitter. Twitter has made it easier for individuals and groups to communicate and share opinions on a

variety of subjects. Genuine Twitter users are seriously at risk from Twitter bots. When Twitter is compromised, rogue accounts may be created and used to execute widespread attacks and deceitful operations. The existing approaches leverage the user's tweets to make this determination, placing more emphasis on accuracy than efficiency. In this study, a feature engineering pipeline has been created to effectively distinguish between Twitter bots and actual users using user metadata such as default name, description, etc. Discussion of several categorization methods, such as neural networks.A comparison of accuracies of all the different Algorithms is also done.
The authors of this paper can conclude that the Decision tree classifier gives the highest accuracy.
As a part of the future scope, this technique can be tested on multiple varieties of data sets. Also, feature engineering can be improved by selecting better features that will give more accurate results, according to the dataset. This technique can also extend to other social media platforms like Instagram, Facebook, etc., where the bots pose a serious threat to genuine users.

VI. REFERENCES

[1]. (2022)The Wikipedia Website-Twitter [Online]. Available:https://en.wikipedia.org/wiki/Twitter

[2]. (2022) Leading countries based on number of Twitter users as of January 2022 [Online].Available:https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/

[3]. (2022) Twitter Advantages and Disadvantages | How Twitter is Used?, Pros and Cons of Twitter [Online].Available: https://www.aplustopper.com/twitter-advantages-and-disadvantages/

[4]. Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. In Proceedings of the 26th International Conference on World Wide Web Companion (WWW '17 Companion). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 963–972. https://doi.org/10.1145/3041021.3055135DataSet - http://mib.projects.iit.cnr.it/dataset.html

[5]. Ranjana Battur, Nagaratna Yaligar, "Twitter Bot Detection using Machine Learning Algorithms", International Journal of Science and Research (IJSR), https://www.ijsr.net/get_abstract.php?paper_id=ART20199245, Volume 8 Issue 7, July 2019, 304 - 307, #ijsrnet

[6]. A. Derhab, R. Alawwad, K. Dehwah, N. Tariq, F. A. Khan and J. Al-Muhtadi, "Tweet-Based Bot Detection Using Big Data Analytics," in IEEE Access, vol. 9, pp. 65988-66005, 2021, doi: 10.1109/ACCESS.2021.3074953.

[7]. Kabakus, Abdullah Talha & Kara, Resul. (2017). A Survey of Spam Detection Methods on Twitter. International Journal of Advanced Computer Science and Applications. 8. 10.14569/IJACSA.2017.080305.

[8]. (2019) Twitter Bot Detection by Nivranshu Pasricha, Conor Hayes . [Online]. Available: https://paperswithcode.com/task/twitter-bot-detection#task-home

[9]. Loukas Ilias, Ioanna Roussaki,Detecting malicious activity in Twitter using deep learning techniques,Applied Soft Computing,Volume 107,2021,107360,ISSN 1568-4946,https://doi.org/10.1016/j.asoc.2021.107360. (https://www.sciencedirect.com/science/article/pii/S1568494621002830)

[10]. M. Fazil and M. Abulaish, "A Hybrid Approach for Detecting Automated Spammers in Twitter," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 11, pp. 2707-2719, Nov. 2018, doi: 10.1109/TIFS.2018.2825958.

[11]. A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu and C. Yang, "CATS: Characterizing automation of Twitter spammers," 2013 Fifth International Conference on Communication Systems and Networks (COMSNETS), 2013, pp. 1-10, doi: 10.1109/COMSNETS.2013.6465541.

[12]. (2021) The IJARCCE website. [Online]. Available:https://ijarcce.com/papers/twitter-bot-detection/

[13]. (2002) The Cornell University website. [Online].Available:https://arxiv.org/abs/2002.01336

[14]. N. Narayan, "Twitter Bot Detection using Machine Learning Algorithms," 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2021, pp. 1-4, doi: 10.1109/ICECCT52121.2021.9616841

[15]. (2022) Correlation (Pearson, Kendall, Spearman)[Online] Available-https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/correlation-pearson-kendall-spearman/

[16]. (2022) Decision Tree Classification Algorithm[Online].Available:https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm

[17]. (2022) Decision Tree Classification Algorithm[Online].Available:https://scikit-learn.org/stable/modules/tree.html

[18]. (2022) multinomial-naive-bayes [Online]. Available:https://www.upgrad.com/blog/multinomial-naive-bayes-explained/

[19]. (2022) Applying Multinomial Naive Bayes to NLP Problems [Online].Available:https://www.geeksforgeeks.org/applying-multinomial-naive-bayes-to-nlp-problems/

[20]. (2022) Classification Algorithms - Logistic Regression [Online].Available:https://www.tutorialspoint.com/machine_learning_with_python/classification_algorithms_logistic_regression.htm#:~:text=Logistic%20regression%20is%20a%20supervised,be%20only%20two%20possible%20classes.

[21]. (2022) Classification Algorithms - Logistic Regression [Online].Available:https://www.geeksforgeeks.org/understanding-logistic-regression/

[22]. (2022) Random Forest Algorithm [Online].Available:https://www.javatpoint.com/machine-learning-random-forest-algorithm

[23]. (2022) Random Forest Classifier [Online].Available:https://www.sciencedirect.com/topics/computer-science/random-forest-classifier

[24]. (2022) K-Nearest Neighbor(KNN) Algorithm for Machine Learning [Online].Available:https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

[25]. (2022) A Guide to AdaBoost: Boosting To Save The Day [Online] .Available:https://blog.paperspace.com/adaboost-optimizer/

[26]. (2022) A Guide to AdaBoost: Boosting To Save The Day [Online] .Available:https://www.geeksforgeeks.org/ml-voting-classifier-using-sklearn/