Detection of fake identities on twitter using supervised machine learning

by

MD. Takiur Rahman
15101120
Ataul Mim Likhon
15101096
A. S. M Musfiqur Rahman
14301094
Mihadul H. Choudhury
19141038

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2019

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

MD. Takiur Rahman
15101120

Ataul Mim Likhon
15101096

A. S. M Musfiqur Rahman
14301094

Mihadul H. Choudhury
19141038

Approval

The thesis titled “Detection of fake identities on twitter using supervised machine learning” submitted by

1. MD. Takiur Rahman (15101120)
2. Ataul Mim Likhon (15101096)
3. A. S. M Musfiqur Rahman (14301094)
4. Mihadul H. Choudhury (19141038)

Of Summer, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 16th, 2019.

Examining Committee:

Supervisor:
(Member)

Hossain Arif
Assistant Professor
Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Md. Abdul Mottalib, PhD
Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Social media has changed the way people get their news. Once people used to buy newspapers to get their news but now everything is online. It has changed the dimension of how we receive news altogether. With a growing effect of social media, it brought us the good, bad and ugly as social media is filled with spams and hate speech along with fake news. One of the crucial problems of social media is fake accounts.We planned to get rid of all fake accounts using machine learning specifically Artificial Neural Network model. Our purpose was to filter out fake accounts from all the accounts existing on social media. There has been a lot of work on this subject, though no permanent solution could be found. We have collected data from many sources and used around four classifiers to compare and determine which is the best classifier for our paper. We have used numeric attributes from twitter accounts and based on these attributes we were able to find out fake accounts. We gave more priority to Artificial Neural Network as we can give different weights to different attributes and get a more accurate result. Also, we are using K-Nearest Neighbor, Random Forest, Support Vector Machine and Neural Networks to compare between the algorithms.Twitter is known for their fake account problem as the user base of Twitter has just grown with time so has the fake users. So, our paper is based on how we can detect this fake account and bots along with making Twitter more safer for its users.

Keywords: Data Mining; Machine Learning; Fake Account; Twitter; Bots;

Dedication

I would like to dedicate this thesis to my loving parents

Acknowledgement

We would like to express our gratitude to the Almighty who gave us the opportunity, determination, strength and intelligence to complete our work.

We would like to thank our supervisor Mr. Hossain Arif for not giving up on us when things become tough and helping us cope with problems and always coming up with more innovative and clear idea to help us finish our paper in time. Also we would like to thank our co supervisor Ms. Najeefa Nikhat Chowdhury for showing us correct paths and giving us good advises.

Lastly, our gratitude goes to the faculty members of the Department of Computer Science and Engineering, BRAC University from whom we gained the knowledge, appreciation and help for the completion of our thesis work.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this introduction, we will discuss different possibilities and ways to detect fake identities on social media platforms. We also talked about three classifiers we used on our research for better accuracy and precision.

1.1 Introduction

Since the development of the Internet in the present time, the number of clients and its applications has expanded to such a degree, to the point that Internet has been recognized as a crucial right in different parts of the world. With this increase in usage, social networking platforms have become the primary channel for all celebrities and organizations to reach their followers regarding their products, publicity and marketing purposes. This phenomenon has risen rapidly throughout the last twenty years. As of late, social networking stages have turned out to be progressively well known not just for people expecting to stay in contact with their associates effectively over the Internet, yet in addition for big names who consistently post and refresh their data utilizing their records to keep every one of their fans refreshed about their exercises. Then again, they experience the ill effects of extending the number of fake accounts that have been made.

Fake accounts imply that the records don't have a place with genuine people. These accounts can exhibit counterfeit news, deceiving web rating, and spam. Fake accounts disregard Twitter Rules. They act in a restricted manner. It can be computerized record interactions or endeavors to beguile or misdirect individuals, for instance, posting unsafe links, forceful behaviors like too much following or not following, making numerous accounts, presenting over and again on ripping points or copy refreshes, posting links with unrelated tweets.

Since the last quarter of 2018, Facebook had 2.27 billion month to month dynamic clients. Then more networking platforms like Twitter have in excess of 336 million clients, and out of these statistics, there is so many accounts which is fake that means it has fake details, name, photographs and other information. Fake follower accounts are not endured by most Social Networking stages and are as a rule effectively restricted once found. Though Online social networks have attracted several malicious activities and research community has offered a number of solutions in the past to the problem. To identify the fake accounts there had been many processes involved. A few researchers used classifiers like Naïve Bayes, a few used a new algorithm called Automated Feature-based profile detection algorithm, then some used machine learning techniques and many more.

In our research, at first, we only focused on the concept of canvas fingerprinting process and graph centrality measures for our experiment. But to implement these two techniques into our research we have encountered a few problems. One of the major problems was finding accurate data sets. We have researched and get in touch with some researchers who had already done their research regarding this topic to find a dataset that suits our project. Though we have found some still we were not happy with the possible solutions. Therefore, we diverged from canvas fingerprint process and graph centrality measures and readjust our center of attention on different classifiers and algorithms such as random forest, decision tree classifier Artificial Neural Network for a better implementation.

Chapter 2

Related Work

This chapter contains literature review, related work with machine learning. Also, this chapter focused on previous works and research related to detection of fake accounts. Additionally, this chapter will guide regarding our research activity.

2.1 Machine Learning

Machine learning is a huge part of artificial intelligence and the study of a system that can learn from data. The world is full of different kind of data generated not only by people but also by computers, phones, and other devices. Pictures, music, words, spreadsheets, videos and it doesn't look like it's going to slow down anytime soon. Machine learning brings a promise of deriving data from all of the data. Machine learning is nothing but rather tools and technology that you can utilize to answer questions from the given data. The biggest example of machine learning is Google search, every time you use Google search, that has machine learning system at its core from understanding the text of your query to adjusting result based on your personal interest.

Machine learning is to create an accurate model that answers our questions correctly most of the time. Firstly, in order to train the model, we need to collect the data to train on. After that, we have to declare the features of data and it is important to have the quality and quantity of data to measure a predictive model. The next step for workflow is choosing a model that researchers and data scientists have created so far. Some suited for image data, for sequence data, for text data, some for numerical data. Though there is a different kind of model, among the main two are:

1. Supervised learning
2. Unsupervised data

After training and testing, the machine learning model will give an answer. Based on the answer further evaluation will be conducted.

2.2 Supervised Learning Model

Supervised learning is the machine learning task of learning capacity. It maps an input to an output dependent on input-output pairs. It induces a function from labeled training data that consist of a lot of training examples. In supervised learning, every precedent is a couple comprising of an input object and the desired output value. It also referred to as the supervisory flag. A supervised learning calculation breaks down the preparation information and produces a construed work, which can be utilized for mapping new models. An ideal situation will take into account the calculation to effectively decide the class names for concealed cases. This requires the taking in the calculation, to sum up from the preparation information to concealed circumstances in a ”sensible” manner.

The parallel undertaking in human and creature brain research is frequently alluded to as concept learning.

2.3 Cross Validation

In cross validation, the main purpose is to select the most effective and best parameters for the algorithms used in the research. It is difficult to track every single data which can cause issues or problems and provide an insufficient result at the end. In this case, cross validation plays a vital role. Using cross validation, these issues and problems can be identified from multiple input features. This process helps to avoid two major problems: Overfitting and Underfitting. It helps us to detect the quality of the model ensuring best performance.

2.4 Related Works Research

Online social networks have attracted several malicious activities and research community has offered a number of solutions in the past to the problem. A few research papers focused on identifying the reason behind having numerous fake accounts on social media. A few scientists utilized example coordinating calculations to identify if there is an example in account names, or if a few records post tweets in a specific time designs. To start with, it may be noted that many individual approaches to solve this issue. A few algorithm is created for assessing confided in relations. A few calculations additionally attempted to take care of this issue in light of social diagram division among client characters. Automated Feature-based profile detection algorithm is also introduced that depend on some machine learning considerations. However, many prominent researches have been stuck with very basic ideas implementations.

2.4.1 Detecting Malicious User Accounts Using Canvas Fingerprint

There are some researches on employing the concept of canvas fingerprinting [16] which works by using the pixel data produced through rendering a text that is drawn on a canvas, then retrieving it back and storing it as a fingerprint of the user. The detection stage mainly depends on storing and checking individual stages.The canvas storing stage starts when a user first logs-in and his input is validated and stored in the database. If the entered credentials are valid, the web application redirects the user to the canvas fingerprint storing page. This data is hashed by any secure hash function and stored in the database as a fingerprint. An enormous web technology which is termed HTML5 is used along with some supporting text rendering such as CSS3 [13]. All these features make canvas fingerprint a good choice to depend on for detecting if several accounts belong to the same user. The reason behind this process works as a fingerprint is that the similar text or picture can be rendered into different ways on different devices based on several factors. These factors basically includes the operating system type and version, font library installed in the device, its graphics card, graphics driver and the browser. Afterwards, the web application redirects the user to the normal default page the user reaches normally after creating a new account. If several accounts is connected to a single fingerprint, it gives a crystal clear indication that these account holders are fake users.

2.4.2 Fake Accounts Detection on Twitter Using Blacklist

Author Myo Myo Swe and Nyein Nyein Myo on their paper used blacklist method for detecting fake accounts on social media . They have used 500 fake words to detect those fake account on twitter. Their whole process is based on content they search and flag specific words to detect if someone is showing traits of fake account . They have used multiple datasets. In one of their data set, there are 1065 users of those user 355 are spammers and 710 are legitimate users. Their accuracy level was around 87.4% on support vector machine classifier. They used around four classifiers and for the test the accuracy level was around 95.7%. Their approach is far more better than the traditional approach word list. We will definitely takes some help from this paper [19]. Author Chakroborty in his paper used 20 features and svm gave the best accuracy [12] .

2.4.3 Twitter Fake Account Detection (Buket ar oslem)

In this research [17] , they processed their data set using supervised discretization technique named EMD (Entropy minimization discretization) on numerical features and analyzed the results with naïve Bayes algorithm [14] .They collected data manually and briefly investigated along with the result from three different individual, common decisions are selected and put in the dataset. They made a dataset which contains 501 fake and 499 real account as they are paying more attention to balance the number of data for the sake of the quality of the results. They took 16 features from them 13 features taken from the information of twitter API. Class decisions are made by username, background image, profile image, followers and friend count number of tweets and content of tweets. Another 3 features added by the researcher like urls_average, mentions _average, hashtags _average. After preparing the dataset they applied naive Bayes learning algorithms without discretization, as a result of first experiment 861 of 1000 instances are classified correctly with the 86.1% accuracy. 112 of 501 fake accounts are classified as real and 27 of 499 real accounts are classified as fake. Weight average of the F measure is 0,860.Secondly they used EMD [16] with minimum description length is applied on numeric features and as a result 901 of the 1000 instances are classified correctly with the 90.9% accuracy.60 of 501 fake account are classified as real and 31 of 499 real accounts are classified fake .Weighted average of the F-measure is increased to 0,909. we have had so much help from this paper and we have used parts from this paper in our paper.[11]

2.4.4 Machine Learning to Detect Fake Identities: Bots vs Humans

In this research, a supervised data is used having different features such as Name, Follower Count, Language, Location, Profile Image, Timezone etc. These features are mostly used by machine learning models referred as engineered factors . Features are divided into three different groups: Data describing the identity of the account, The relationship of the account to others and The behavior of the account [20]. They focused on frequency of messages and time of day based on individual accounts to create a comparison between malicious accounts and original accounts. They have used light weight classifiers which only includes data describing the identity of an individual account. In case of Datasets, the authors tried a different way to manage appropriate datasets. They created deceptive accounts in order to inject it manually in the existing corpus. They formed these deceptive accounts with the help of past psychological researches. They filtered the data based on some psychological evaluations such as people usually lie about their age, location, gender, occupation, educational background [1] along with their names. These information helped them to create a set of informed deceptive accounts. The authors inject fictitious accounts into original corpus. They used supervised machine learning such as random forest boosting and cross validate the data. Using this procedure, they came up with the detection of fake accounts on twitter. The accuracy level using this method was 49.75%.

2.4.5 Detect fake accounts using Graph Centrality measures

This research tried to find the fake account using machine learning algorithms.While researching we found some limitations like availability of dataset, labeling data set, algorithms we should go with, connecting dataset with algorithms. The objective of this research is to come up with a possible solution to detect fake account in twitter with a maximum result using some machine learning algorithms.While experiment researchers observed in the data set many nodes with edges in the last sub-graph file are accounts which are friends and followers listed in the user.csv are not labeled. For the solution they removed some of the edges which are connected with one node. This solution gave them a clear view of all nodes and reduced the size of the dataset and scheme resource requirement for the processing purpose. As this is a graph centrality based research, After labeling [5] the data set they tried to find some centralities used for classification like as Betweenness Centrality, Eigenvector Centrality, In-degree Centrality, Out-degree Centrality, Katz Centrality, Load Centrality these centrality measures can be computed using Python library,citehagberg2008exploring. NetworkX returns all kind of required centrality measures that generate a labeled users.csv file. Filtering the data set they found 1920 records that contain centrality based feature.[15] Once the data set is ready, they divided them into two segment, one is for training the classifier consisted 75%

of the dataset and the other one is for testing the accuracy of classifiers consisted of 25% of the dataset.

Table 2.1: Distribution of Training and Testing Sets

Dataset	Number of Records	Legitimate Followers	Fake followers
Training set	1440	936	504
Testing set	480	286	194

For experiment they trained and tested their approach on ANN, Decision tree, Random forest classifiers. Final result was obtained using Artificial Neural Network, Random Forest Some other classifiers based on accuracy level.

2.4.6 Detection of Fake Followers using Feature Ratio in Self-Organizing Maps

StatusPeople and Social Banker .They have used The Fake Project Dataset for the analysis that are presented in that paper. They have used mainly two user data sets. One is E13(elections 2013) that consisted 100% humans and another one is FSF(fast followers) that consists 100% fake followers. So from the datasets they proposed to use 4 different fields with numerical values such as Follower count, Friend Count, Status Count and Favourite Count. In order to identify outliers in the dataset that characterize possible fake accounts a feature ratio is introduced in their proposed strategy. Of the above four factors, two are used to obtain the proposed feature ratio which is subsequently used as the input variable by the neural network [7] for clustering. The way they have calculated feature ratio from these is, FeatureRatio = FavouriteCount/StatusCount. Finally, they have used clustering with and without feature ratios for the purpose of identifying fake behavior from the dataset, they run the given experiment with the four identified factors as input and also with the proposed feature ratio. Moreover, they ascertained that using the factors independently does not help in isolating fake accounts effectively. The proposed metric (FR), demonstrates a clear offset between a genuine and fake user thereby establishing that the feature ratio is effective measure for fake behavior. [18] During our thesis, we researched on several topics and discussed ideas in order to come up with the best solution. We took effective ways and knowledges from previous works and updated our thesis with a new era of research. In our thesis, we will be using machine learning algorithms such as ANN, Decision Tree classifiers etc to detect fake accounts on social media.

Chapter 3

System Implementation

Our first task for this research was to collect dataset. We found a researcher on researchgate who was willing to share his dataset with us. Thus, we emailed him and got access to the datasets. We then started figuring out which attributes are compatible with our paper and algorithm and then we started to delete the attributes that were not necessary and thus we ended up with around 5 optimal attributes that are crucial when it comes determining if the user is fake or not. We have divided the dataset into two parts one for training and another one for testing. we first applied k neighbor algorithm, random forest, svn, artificial neural network on the testing set. After a while, we used the main data set and used this same algorithm and found the accuracy, f-measure, false positive, false negative.

3.1 Proposed Model

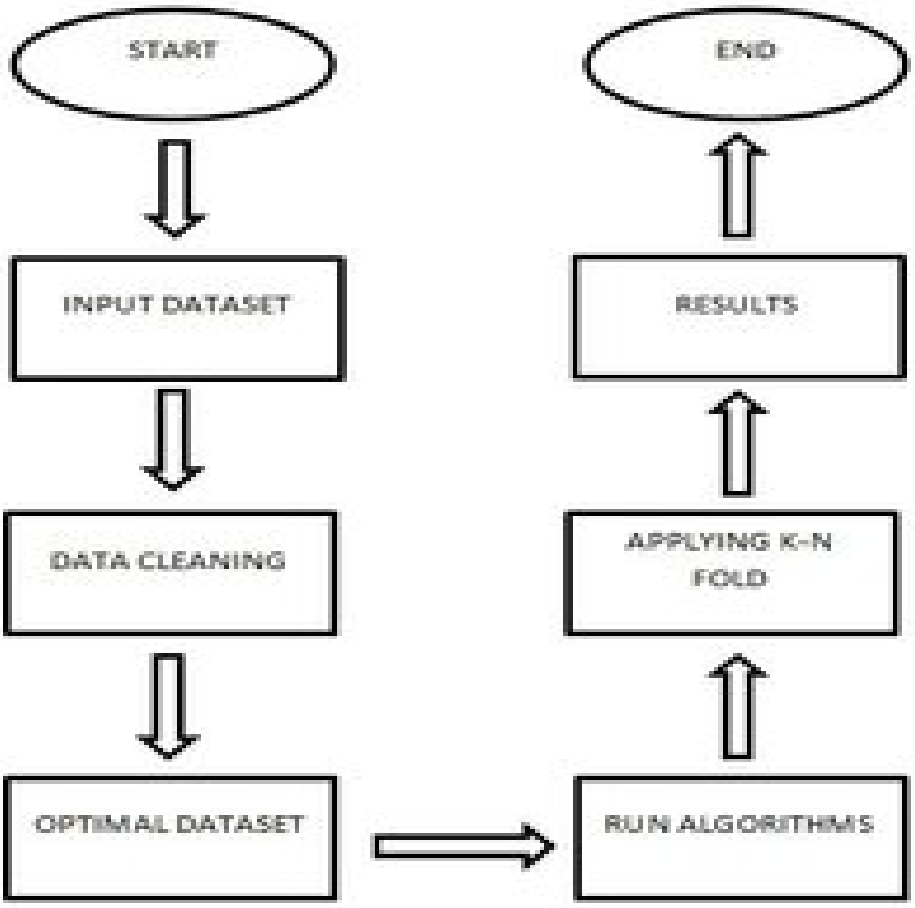


Figure 3.1: Overview of our whole system

3.2 Implementation

3.2.1 Data Collection

We were looking for the dataset for about 3 months before finding our first set of the dataset from GitHub it had about 3000 instances and about 16 attributes. But after some searching and looking online we came across a big data set name (@fakeproject) and it had more instances and features, but our final dataset contained about 6825 instances so much bigger than the previous one. so it was perfect for our paper and this data set contained all the necessary information of a Twitter user, So our job became quite easy after that also.

3.2.2 Data Description

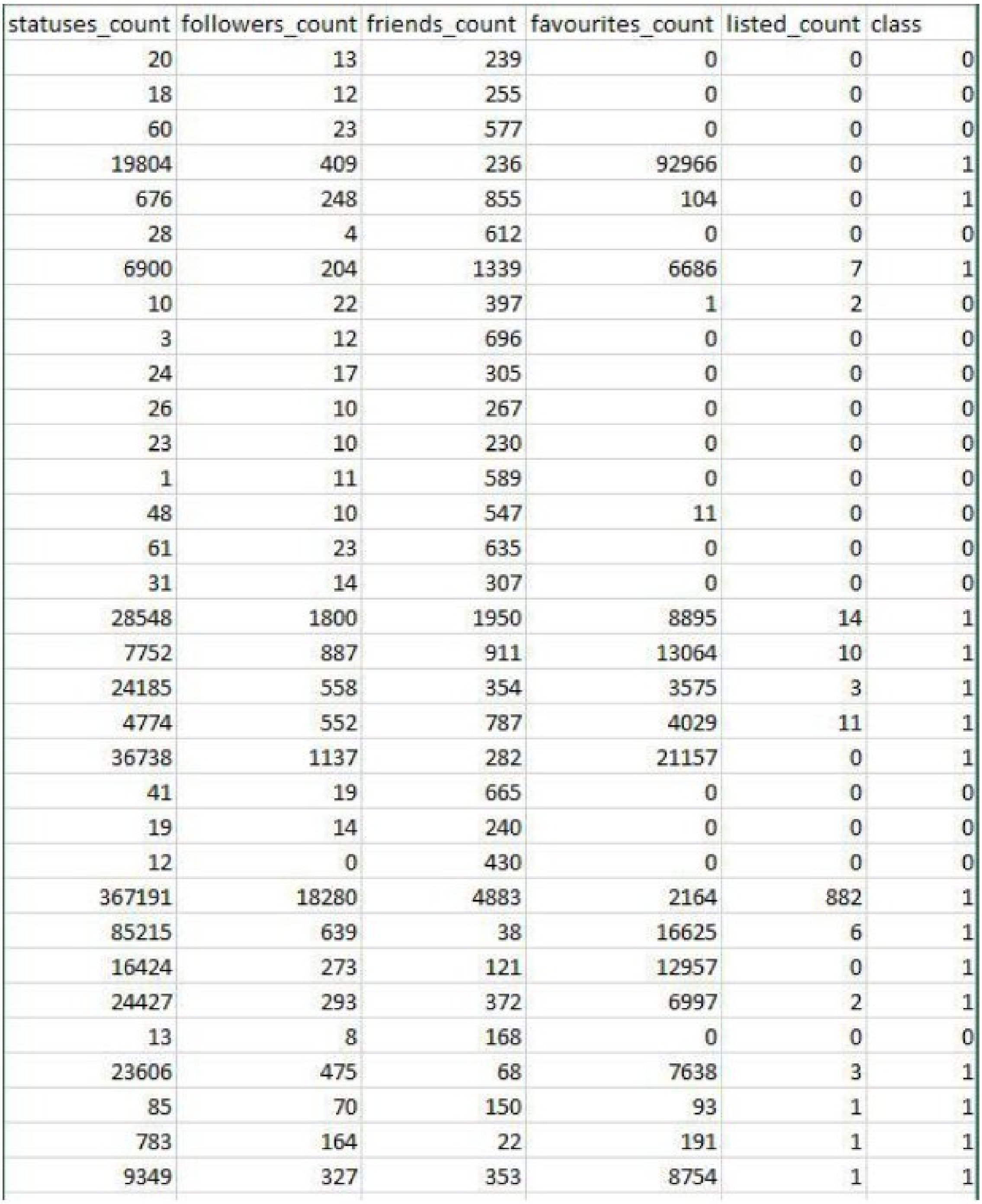
Our data has around 16 features but majority of the features could not be used as some of them or text based or some of them or biased towards the result. During the first months of our research we had chosen over 10 features with time we had let go of many features. There was default picture attributes the fake accounts almost all of them had default profile picture so we thought it would not do justification to our research by including . the fake accounts had more default photo, background default photo, default color schemes so we thought it would be better to not include those attributes as that you would give inaccurate result as these attributes would start to dictate the results. We have put a table of our first set of features.

Table 3.1: Table of sample data attributes and description

Attributes	Description
Name	Name of the account holder
Screen name	Twitter handle
Follower count	How many followers the account has
Following count	How many people the account follows
Created time	Timestamp of when the account was created
language	What is the chosen
timezone	In which time zone the account recides
location	The location of the account holder
Profile image	Image of the account holder
Verified	If the account is verified from twitter
Profile description	The bio on twitter
Color scheme	If the user has chosen a custom color
Tweet count	How many tweets have been tweeted

3.2.3 Data Visualization of the Final Data

Data visualization is very important as we can see the pattern of the data from this process. Also, It helps people to understand the data as it's just not texts. Furthermore , We can find the attributes that decide which one is important in making the decisions , also with data visualization we can find which attributes are very similar and contribute to a bad results. We are using graph to show you data .Although, there are many ways data can be visualized . we choose this way as it would be easier for people to understand what data we collected. The blue one means “Real” and the red one means “Fake” We have put the question on the heading and on X-axis we have divided the data based on the question and there are two answers yes or no. so we can find the every possible result



statuses_count	followers_count	friends_count	favourites_count	listed_count	class
20	13	239	0	0	0
18	12	255	0	0	0
60	23	577	0	0	0
19804	409	236	92966	0	1
676	248	855	104	0	1
28	4	612	0	0	0
6900	204	1339	6686	7	1
10	22	397	1	2	0
3	12	696	0	0	0
24	17	305	0	0	0
26	10	267	0	0	0
23	10	230	0	0	0
1	11	589	0	0	0
48	10	547	11	0	0
61	23	635	0	0	0
31	14	307	0	0	0
28548	1800	1950	8895	14	1
7752	887	911	13064	10	1
24185	558	354	3575	3	1
4774	552	787	4029	11	1
36738	1137	282	21157	0	1
41	19	665	0	0	0
19	14	240	0	0	0
12	0	430	0	0	0
367191	18280	4883	2164	882	1
85215	639	38	16625	6	1
16424	273	121	12957	0	1
24427	293	372	6997	2	1
13	8	168	0	0	0
23606	475	68	7638	3	1
85	70	150	93	1	1
783	164	22	191	1	1
9349	327	353	8754	1	1

Figure 3.2: Sample of column headers of our final data

1. statuses count = it means how many statuses the user has put out
2. followers count = it means how many followers the user has
3. friends count= it means how many people the user in question is following
4. favourites$_{c}ount = itmeanshowmanystatusesortweetstheuserhasliked$
5. listed$_{c}ount = meansiftheuserhasbeenaddedtoanylists$
6. class= 0 means fake account and 1 means real account

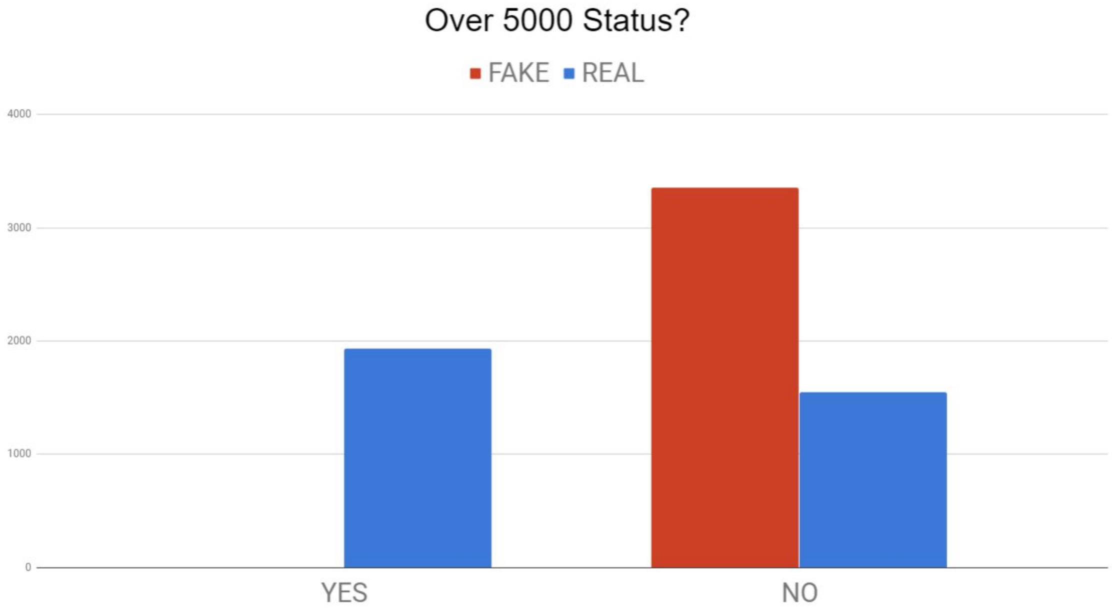


Figure 3.3: Data comparison for status count of the users

From figure 3.1, we can see we have more number of user who are fake if they put out less than 5000 statuses from about 6825 users user who have more than 5000 statuses and are real are 1934 and of them fake are only 4 . So there's a correlation between having put out less status and being a fake account .Furthermore, it's the same story in the users who have put out less than 5000 statuses of those user majority are fake around 3350 users are fake and 1547 users are real .

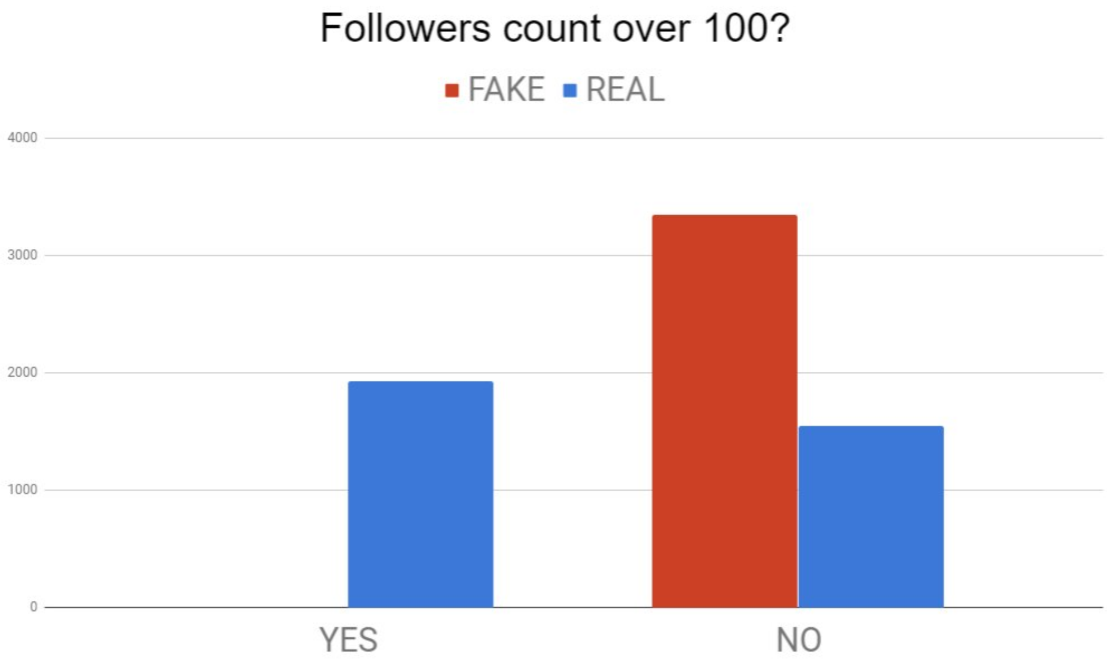


Figure 3.4: Data comparison for Followers count of the users

From figure 3.2, we can see we that followers count is a very important attribute as fake users usually don't have many followers as people only follow people they know or people who are famous so it's a red flag when it comes to people with fewer followers. Form the dataset only 46 users have more than 100 followers. The real users who have more than 100 followers are around 2876. The user who have less than 100 followers only 598 are real users and 3305 are fake users. So we have come to the conclusion that this attribute is very important when it comes to finding out if the users are fake or real and people with less follower are more likely to be fake.

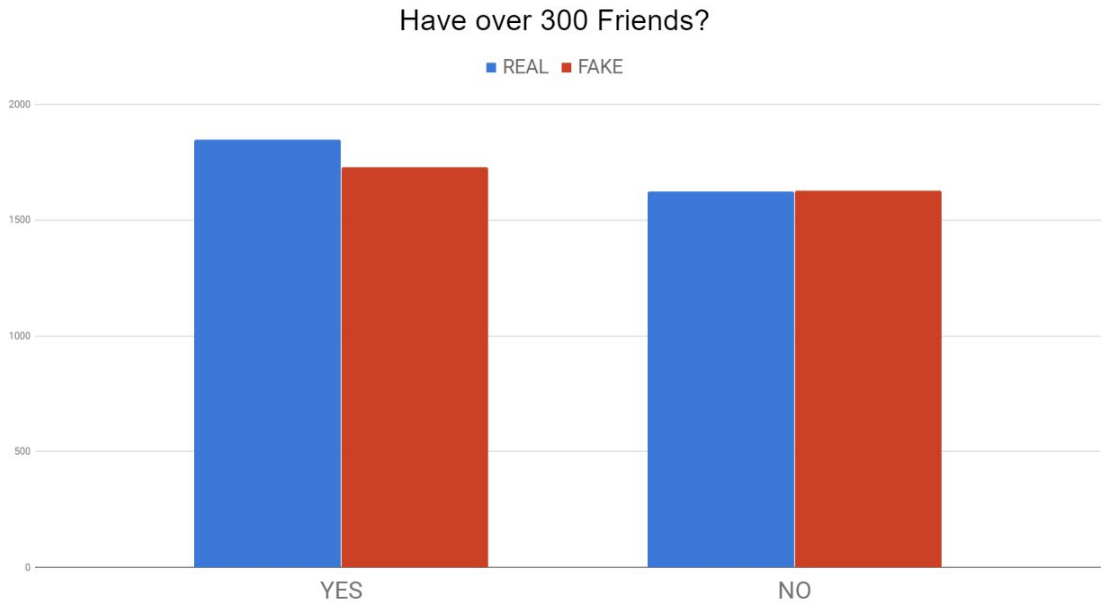


Figure 3.5: Data comparison for Friends count of the users

The data from friends count attributes are hard to differentiate as the ratios of these data are quite similar. so this is maybe not the most optimal attribute as we can not come to any conclusion from these attributes. Fake users may follow a lot of users or they may follow less user to and have more statuses. So it is harder to tell if the users are fake or real by finding out how many users they follow. Users more than 300 friends and are fake is 1730 and real users are a bit more around 1849 users. The users with less than 300 friends and fake and real are differentiated by 1 user the fake user are 1626 and real user 1625.

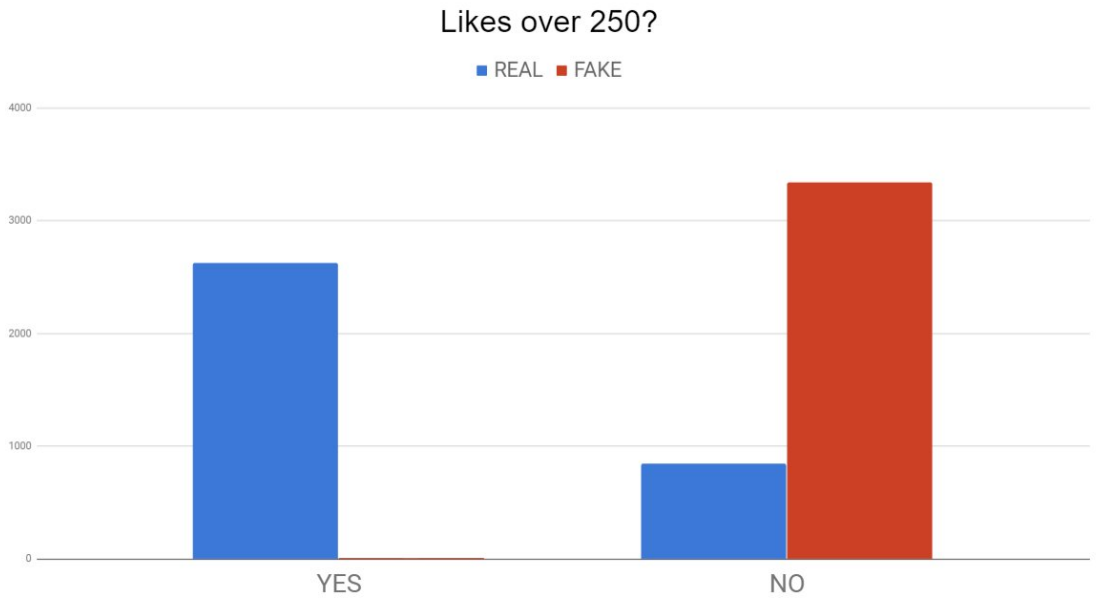


Figure 3.6: Data comparison for Likes count of the users

The active users like statuses and pictures and the fake users tend to have less activity and like less statuses. As we can see the users who have more than 250 likes are mostly real as people who have been using twitter for a long time will obviously have more likes . 2625 number of users are real who have given out more than 250 likes and only 12 people are fake .However it is a different story when it comes to user who have liked less than 250 post only 849 users are real and 3339 users are fake . so we can come to the conclusion that if a user has liked less post there is a chance that she or he may be a fake user

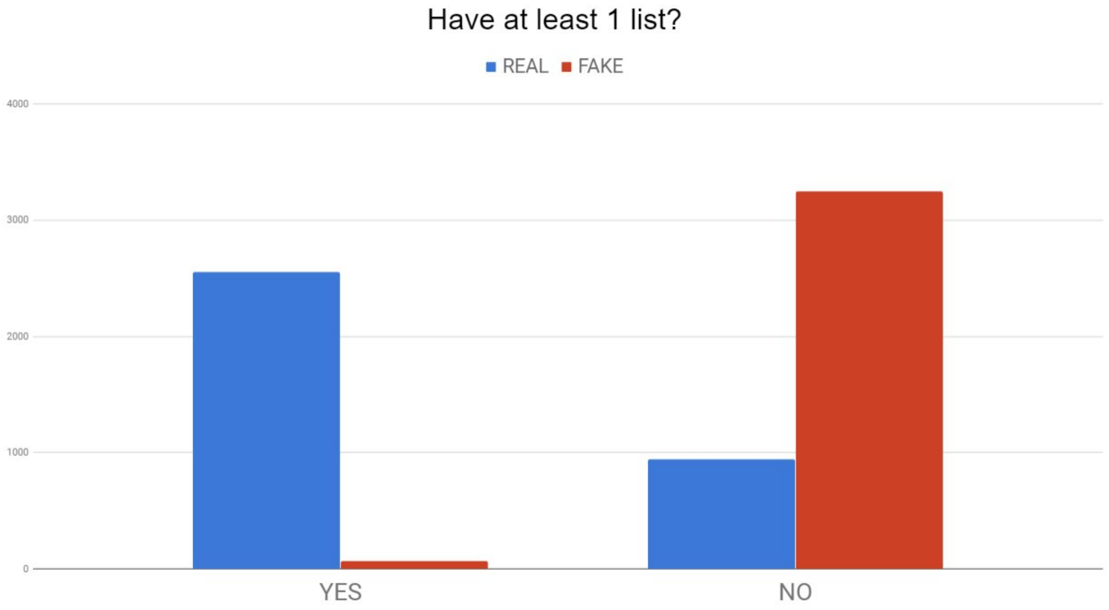


Figure 3.7: Data comparison for listed count of the users

Lists are mainly used to put users in specific group. For fake users it is harder to get into a group as people tend put people they know on their lists. So only 70 fake user have been included in a list and around 2557 user have at least 1 list . The user who don't have a single list of them around 3250 users are fake and 944 users are real

Chapter 4

Algorithms Used

4.1 Support Vector Machine (SVM)

Support vector machine is concepts of decision planes that defines the boundary of a decision. Objective of the support vector machine is to find a hyperplane in the number of features that distinctly classifies the data point. It is primarily a classier method that performs tasks in a multidimensional space that differentiate cases of different class label by constructing hyperplane. SVM can handle multiple continuous and different categorical variables. As well as SVM supports both regression and classification. Here the illustration shows the basic concept of SVM. We see the original object on the left side of the schema are arranged by some set of mathematical function. This rearranging process is called mapping[8], Here the arranged objects are at the right side of the schema. Beside the simple linear case, there would be many different cases like polynomials, splines, radial basis function networks, multilayer perceptions could be come through SVM operation.

Algorithm for SVM
Initialize: n-dimensional hyperplane, where, n number of features learning rate. Repeat until the closest point on all axes is furthest from the hyperplane:

$$\text{maximize } f(c_1 \ldots c_n) = \textstyle\sum_1^n c^i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n y_i c_i (x_i . x_j) y_j c_j,$$
$$\text{subject to } \textstyle\sum_{i=1}^n c_i y_i = 0 \text{ and } 0 \leq c_i \leq \frac{1}{2n\lambda} \tag{4.1}$$

4.2 Neural Network

Neural network is a role model of inspired by the structure of human brain. Human brain is a collection of cells known as neurons, when we see something familiar to us some portion of neurons light up. These neurons are inter-connected to each other and always try to communicate with another to come up with a result with most accuracy.

Just like human brains researchers create neural network. Nodes are connected to each other, sharing their resources to find the most accurate result, updating the perception result. Because of the connection, it is also known as connectionist computer network passing internal values to each other . Basic Sigmoid function [6] for neural network is –

$$a = \sigma(z) = \frac{1}{1 + e^{-z}}$$

(4.2)

Though this sigmoid function[3] works slower but we are using this function. Because it represents as a function itself after the derivation of sigmoid and also helps to implement back propagation for the need of neural network.

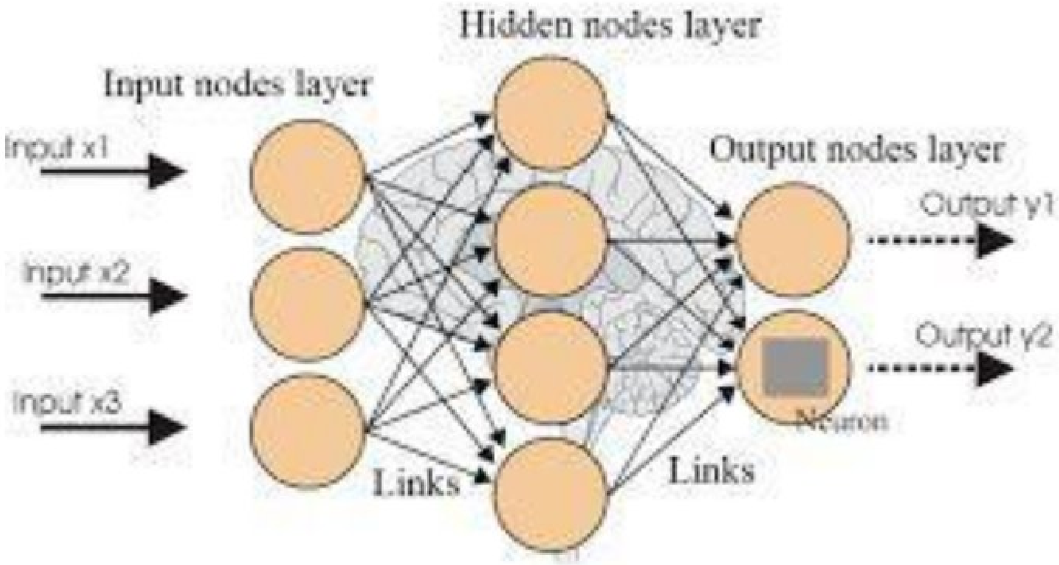


Figure 4.1: Neural Network

As a signal flow works in neural network, at the Starting nodes are for taking inputs and generating an output based on the degree of derivation, weights inside the network. For better output weights inside the network are changed. The layers used in our research are:

Input layer: As the input we uploaded our data for training and labeling the data set. We erase the inefficient data.

Output layer: In this layer we get the output of real profiles and fake profiles. Accuracy level of our data set.

Hidden layer:In this portion neural network learn itself about the dataset to generate output.

4.3 K-Nearest Neighbours

K-Nearest Neighbours is an easy and simple algorithm basically used for classification and pattern recognition. It stores all available cases and generate new cases after classifications. It is one sort of lazy learning where the functions are used locally. The computation process differentiates until the classification period. In machine learning, KNN [2] is one of the simplest algorithm. KNN predicts the new

instances using a searching method where the searching is held throughout the entire training set. It usually avoid training data points for generalization process. So, we can say, the training phase is very efficient and fast. It also has a high accuracy and versatile. Moreover, high memory is required as it stores all training data. For this reason, computation is pretty expensive.

4.4 Random Forest

Random Forest is a mostly used and supervised learning [4] algorithm.Basically it is an ensemble algorithm based on decision tree predictors devised by Breiman, largely influenced by the Bagging method [9]. A general idea of the bagging method, is a combination of learning models increases the overall result. In a simple word random forest builds multiple decision trees and merges them together to get a more accurate stable prediction.

Random Forest has nearly the same hyper parameters as a decision tree or a bagging classifier. Fortunately, you don't have to combine a decision tree with a bagging classifier and can just easily use the classifier-class of Random Forest. One big advantage of random forest is, that it can be used for both classification and regression problems, which form the majority of current machine learning systems.

Chapter 5

Result and Analysis

In the previous chapter, we have discussed the proposed model and our dataset. We went from collecting dataset to cleaning out dataset to getting rid of some of the attributes and then we decided to go with this final attributes. Now we are going to be getting the results from the algorithm, finding the accuracy, f-measure, recall, and precision. We are going to run a different algorithm and find out the results

5.1 Applying Algorithms

Our research work aims to find out if a user is fake or not. Since our data set is numerical and there are two possible results one the user is fake or other the user is real. we have used 4 algorithms for this experiment which area Artificial neural network, Random forest tree, k neighbor algorithm, Support vector machine algorithm. We have found out the accuracy, f-measure, recall, and precision of the algorithm.

5.2 K-Fold Cross Validation

K-Fold cross-validation is used to improve the accuracy [10] of the machine learning model. The problem with the test/train split lead to overfitting. The number of the test set is low compared to train data set which may lead to overfitting. It divides the whole data set in k folds and each fold will contain the same amount of data in it. One fold is selected as a test set and k-1 folds are selected as training set and accuracy of the function is carried out. It is then repeated k times so that every portion of data selected as a test set and training set. As we repeated it k times we get k times mean square error. So the error of this model is computed by taking an average of the mean square error over k folds [42, 81]. It is experimentally found out that setting fold value to 10 gives a result with low biasing. Along with this, it reduces the computation time as it is only 10 Every data point is tested exactly once and trained K-1 times

5.3 Artificial Neural Network

Accuracy is 90.1099% tested on 6825 instances

Table 5.1: Precision and recall of Artificial neural network algorithm on the dataset

	True Fake	True Real	Class precision
pred.Fake	3257	94	84.9
Pred. Real	581	2893	96.9
Class recall	97.2	83.3	

From Table 5.1 we can see, that our accuracy is 90.1099% which is approximately 90%. If we look at the precision, where precision means the total number of true positive in all the prediction of yes. There are 2893 who are predicted as real users and which is true the result came in right . to find the recall on class real we added up 94 and 2893 and found 2987 and then we divided 2893 by 2987 and found the recall of 96.9 and also we found the recall for fake users and so we added up 3257 with 581 which equaled to 3838 and then we divide 3257 by 3838 and got recall percentage of 84.9 . we then found the precision of the results for both the classes first we added the real user class of which there were 3474 users of 2893 are real so precision is 2893 divided by 3474 so the precision is 83.3.Also, we did precision for fake users, totally predicted fake users are 3351 of them only 3257 are actually fake so the precision is about 97.2.

5.4 Random Forest Tree

Accuracy is 99.2088% tested pm 6825 instances

Table 5.2: Precision and recall of random forest tree algorithm on the dataset

	True Fake	True Real	Class precision
pred.Fake	3315	36	99.5
Pred. Real	18	3456	99
Class recall	98.9	99.5	

From Table 5.2 we can see, that our accuracy is 99.2088% which is approximately 99%. If we look at the precision, where precision means the total number of true positive in all the prediction of yes . There are 3456 who are predicted as real users and which is true the result came in right . to find the recall on class real we added up 36 and 3456 and found 3492 and then we divided 3456 by 3492 and found the recall of 98.9% and also we found the recall for fake users and so we added up 3315 with 18 which equaled to 3333 and then we divide 3315 by 3333 and got recall percentage of 99.5%. we then found the precision of the results for both the classes first we added the real user class of which there were 3474 users of 3456 are real so precision is 3456 divided by 3474 so the precision is 99.5% for real user class. Also, we did precision for fake users, totally predicted fake users are 3351 of them only 3315 are actually fake so the precision is about 99.0%.

5.5 K Neighbor Algorithm

Accuracy is 97.8022% tested pm 6825 instances

Table 5.3: Precision and recall of k neighbor algorithm on the dataset

	True Fake	True Real	Class precision
pred.Fake	3298	53	97.1
Pred. Real	97	3377	98.5
Class recall	98.4	97.2	

From Table 5.3 we can see, that our accuracy is 97.8022% which is approximately 97%. If we look at the precision, where precision means the total number of true positive in all the prediction of yes. There are 3377 who are predicted as real users and which is true the result came in right . to find the recall on class real we added up 53 and 3377 and found 3430 and then we divided 3377 by 3430 and found the recall of 98.4% and also we found the recall for fake users and so we added up 3298 with 97 which equaled to 3395 and then we divide 3298 by 3395 and got recall percentage of 97.2% . we then found the precision of the results for both the classes first we added the real user class of which there were 3474 users of 3377 are real so precision is 3377 divided by 3474 so the precision is 97.1% Also, we did precision for fake users, totally predicted fake users are 3351 of them only 3298 are actually fake so the precision is about 98.5%.

5.6 Support Vector Machine Algorithm

Accuracy is 70.315% tested pm 6825 instances

Table 5.4: Precision and recall of Support vector algorithm on the dataset

	True Fake	True Real	Class precision
pred.Fake	3350	1	62.3
Pred. Real	2025	1449	99.9
Class recall	1	41.7	

From Table 5.4 we can see, that our accuracy is 70.315% which is approximately 70%. If we look at the precision, where precision means the total number of true positive in all the prediction of yes . There are 1449 who are predicted as real users and which is true the result came in right . to find the recall on class real we added up 1and 1449 and found 1450 and then we divided 1449 by 1450 and found the recall of 99.9% and also we found the recall for fake users and so we added up 3350 with 2025 which equaled to 5375 and then we divide 3350 by 5375 and got recall percentage of 62.3% . we then found the precision of the results for both the classes first we added the real user class of which there were 3474 users of 1449 are real so precision is 1449 divided by 3474 so the precision is 41.7%. Also, we did precision for fake users, totally predicted fake users are 3351 of them only 3350 are actually fake so the precision is about approximately 1.0

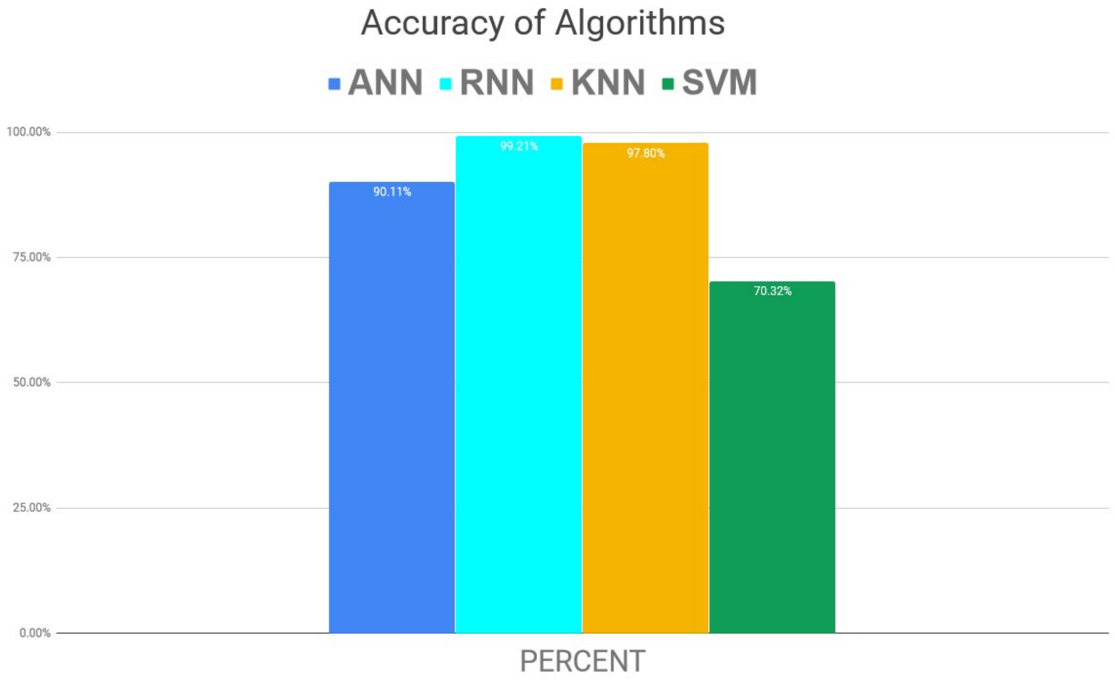


Figure 5.1: Accuracy comparison of between the algorithms

In Figure 5.1, we can see from the accuracy comparison that Random forest tree gives the most accurate result which is around 99%, Highest compared to any of them. The second best algorithm was k neighbor algorithm which gave around 97% accuracy. the artificial neural network gave around 90% which is not bad at all but it was good at detecting fake accounts but when it came to finding real account this algorithm made many mistakes so lots of real users were flagged as fake users. The worst performed algorithm was Support vector Algorithm with 70%, As the dataset was quite clear and had many correlations it should have performed better but this algorithm was very best when it came to finding fake account but when it came to detecting real account this algorithm was around 60% correct so many real users were flagged as fake users.

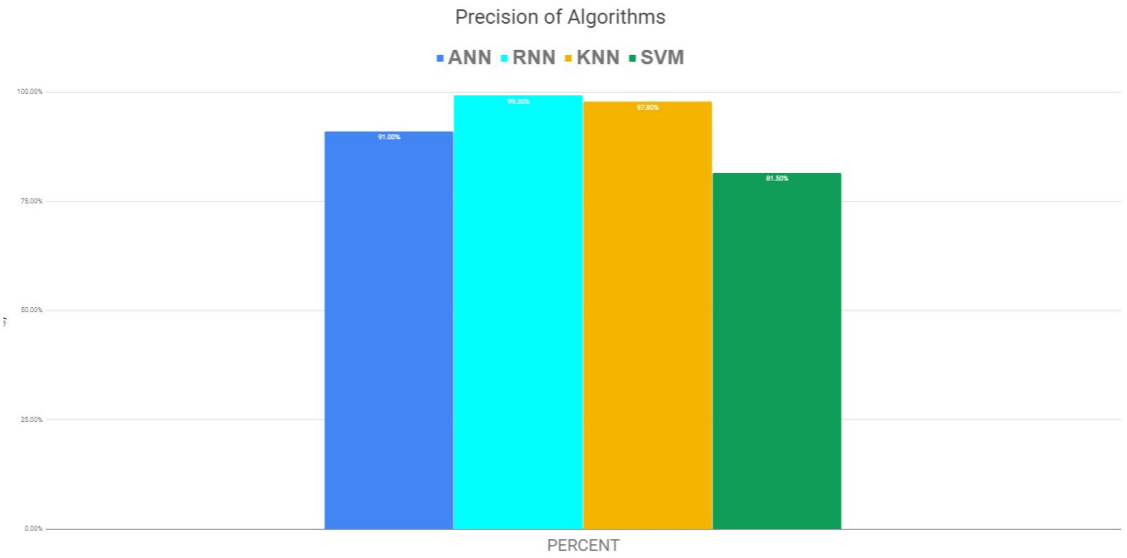


Figure 5.2: Precision comparison of between the algorithms

In figure 5.2 we can see that random forest is still the best-performed algorithm compared to other algorithms, random forest has 99% precision close second was k neighbor with 97.80% precision which is not bad.But , random forest tree clearly is the best algorithm. The last two were artificial neural network and svm with 91% and 81% respectively. Support vector Algorithm was the worst performed the algorithm. The accuracy and precision are quite similar.

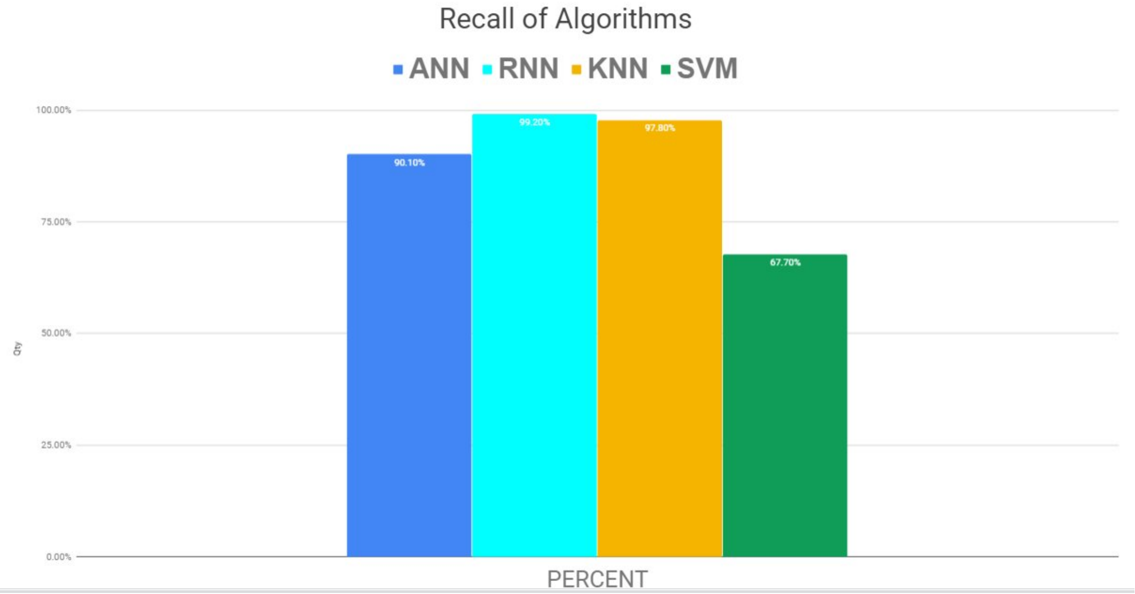


Figure 5.3: Precision comparison of between the algorithms

From figure 5.3 we can, that best-performed algorithm is still random forest tree with 99% recall. And second best is k neighbor with 97.8 so there isn't much difference between the first two.,Both have a very good recall. However, Support vector algorithm performed really poorly with just 67%. and Artificial neural network is still given a stable recall with 90%.

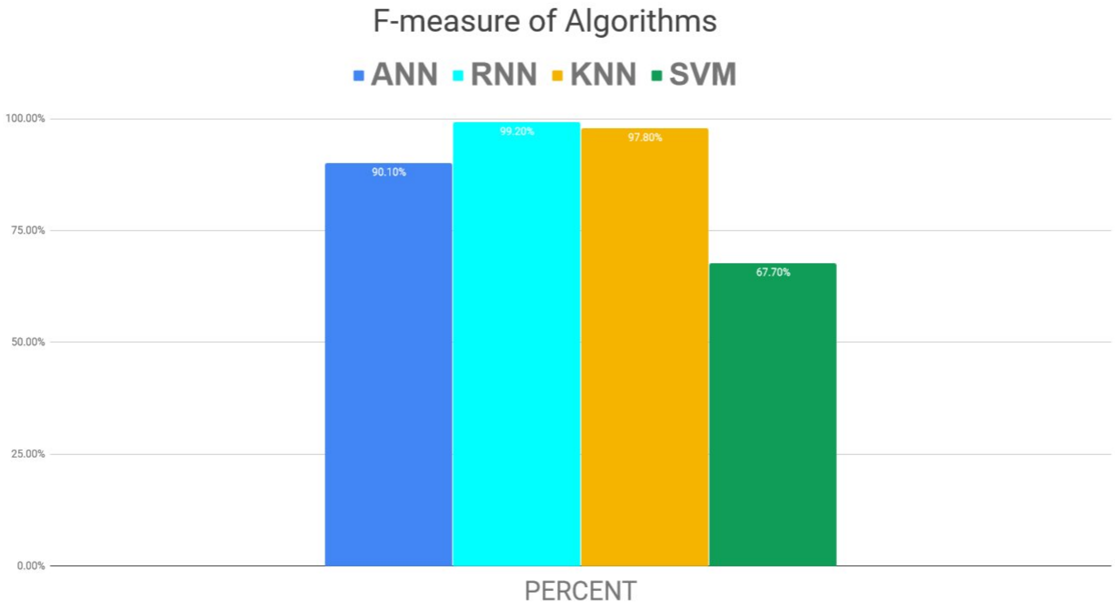


Figure 5.4: F-measure comparison of between the algorithms

In figure 5.4 we can see a familiar trend with figure 4.1, the algorithm which performed best also performed best in F-measure, the f-measure is around 99%, the second best algorithm is k neighbor with 97%. the third and last one is artificial neural network and Support vector Algorithm respectively With 90% and 67%.

Chapter 6

Final Remark

6.1 Conclusion

Our work essentially included recognition of a unique class of followers in social networking platforms known as Fake followers which are being utilized much of the time by famous people and associations to control their fame. While this point has picked up consideration of numerous researchers previously, most scientists have used platform specific features for their classifiers.

6.2 Limitation

- From our experience with data collection process we have found out that data collection is the most crucial step in this research paper as we had continuously email people asking for dataset, At one point we almost gave up on finding data set, Fortunately we found a dataset online via email.

- In our research we understood finding fake account is bit of a guessing game as we could never verify if the user is fake, Also, twitter also bans fake account based on their algorithm and assumption so they are also never sure if the account is fake

- The dataset contains around 7 thousand user but in real life active user is around 326 million so it is harder to scale at that level

Bibliography

[1] R. A. Maier and P. J. Lavrakas, “Lying behavior and evaluation of lies”, *Perceptual and Motor Skills*, vol. 42, no. 2, pp. 575–581, 1976.

[2] S. Dudani, “The distance-weighted k-nearest neighbor rule”, *IEEE trans. on systems, man and cybernetics*, vol. 8, no. 4, pp. 311–313, 1978.

[3] G. Cybenko, “Approximation by superpositions of a sigmoidal function”, *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[4] A. Liaw, M. Wiener, *et al.*, “Classification and regression by randomforest”, *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[5] X. Zhu and Z. Ghahramani, “Learning from labeled and unlabeled data with label propagation”, Citeseer, Tech. Rep., 2002.

[6] X. Yin, J. Goudriaan, E. A. Lantinga, J. Vos, and H. J. Spiertz, “A flexible sigmoid function of determinate growth”, *Annals of botany*, vol. 91, no. 3, pp. 361–371, 2003.

[7] A. Hagberg, P. Swart, and D. S Chult, “Exploring network structure, dynamics, and function using networkx”, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2008.

[8] S. R. Selamat, R. Yusof, and S. Sahib, “Mapping process of digital forensic investigation framework”, *International Journal of Computer Science and Network Security*, vol. 8, no. 10, pp. 163–169, 2008.

[9] T. Chen and J. Ren, “Bagging for gaussian process regression”, *Neurocomputing*, vol. 72, no. 7-9, pp. 1605–1610, 2009.

[10] J. D. Rodriguez, A. Perez, and J. A. Lozano, “Sensitivity analysis of k-fold cross validation in prediction error estimation”, *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 3, pp. 569–575, 2010.

[11] E. Kouloumpis, T. Wilson, and J. Moore, “Twitter sentiment analysis: The good the bad and the omg!”, in *Fifth International AAAI conference on weblogs and social media*, 2011.

[12] A. Chakraborty, J. Sundi, S. Satapathy, *et al.*, “Spam: A framework for social profile abuse monitoring”, *CSE508 report, Stony Brook University, Stony Brook, NY*, 2012.

[13] K. Mowery and H. Shacham, “Pixel perfect: Fingerprinting canvas in html5”, *Proceedings of W2SP*, pp. 1–12, 2012.

[14] T. R. Patil, S. Sherekar, *et al.*, “Performance analysis of naive bayes and j48 classification algorithm for data classification”, *International journal of computer science and applications*, vol. 6, no. 2, pp. 256–261, 2013.

[15] A. Mehrotra, M. Sarreddy, and S. Singh, “Detection of fake twitter followers using graph centrality measures”, in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, 2016, pp. 499–504.

[16] A. Abouollo and S. Almuhammadi, “Detecting malicious user accounts using canvas fingerprint”, in *2017 8th International Conference on Information and Communication Systems (ICICS)*, IEEE, 2017, pp. 358–361.

[17] B. Erşahin, Ö. Aktaş, D. Kılınç, and C. Akyol, “Twitter fake account detection”, in *2017 International Conference on Computer Science and Engineering (UBMK)*, IEEE, 2017, pp. 388–392.

[18] N. T. Simon and S. Elias, “Detection of fake followers using feature ratio in self-organizing maps”, in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, IEEE, 2017, pp. 1–5.

[19] M. M. Swe and N. N. Myo, “Fake accounts detection on twitter using blacklist”, in *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, IEEE, 2018, pp. 562–566.

[20] E. Van Der Walt and J. Eloff, “Using machine learning to detect fake identities: Bots vs humans”, *IEEE Access*, vol. 6, pp. 6540–6549, 2018.