

DETECTION OF FAKE PROFILES USING ANN ALGORITHM

P MANIDEEP REDDY, B SIDDHARTH, R SAIKUMAR REDDY

DEVAVARAPU SREENIVASARAO (Guide)

Department of Computer Science & Engineering

SREENIDHI INSTITUTE OF SCIENCE AND TECHNOLOGY

Yamnapet (V), Ghatkesar (M), Hyderabad –501301, T.S.

ABSTRACT

The unconstrained increase in online social media usage has resulted in unchecked growth in the number of user profiles that are fake and causes a lot of damage to social media networks. This results in a dreadful threat posed to internet security. It is extremely important to identify the fake profiles causing the threat. These user accounts often commit fraud like retrieving personal data and confidential information from the victim accounts, spreading false information about the users online, and threatening and bullying the victims all by making sure that their identities are not revealed to the public. These fraud accounts look realistic but follow some patterns which help identify them. This project aims to help the users identify the difference between genuine users and fake frauds on Facebook. This project uses the concepts of machine Learning particularly Neural networks to identify if the Facebook friend request is genuine or fake. Finally, the solution is found by considering all the important parameters involved in the social network page.

CHAPTER 1 INTRODUCTION

1.1 introduction to fake profiles on socialmedia

As there is huge increase in the usage of social media in recent times there has been also increase in the scope for the cyber-attacks ,the attackers create a social media profiles using the identity details like name, email id and address of the victim without their knowledge and it is called fake profile creation these fraud people create fake profiles in order to harmthe victim.

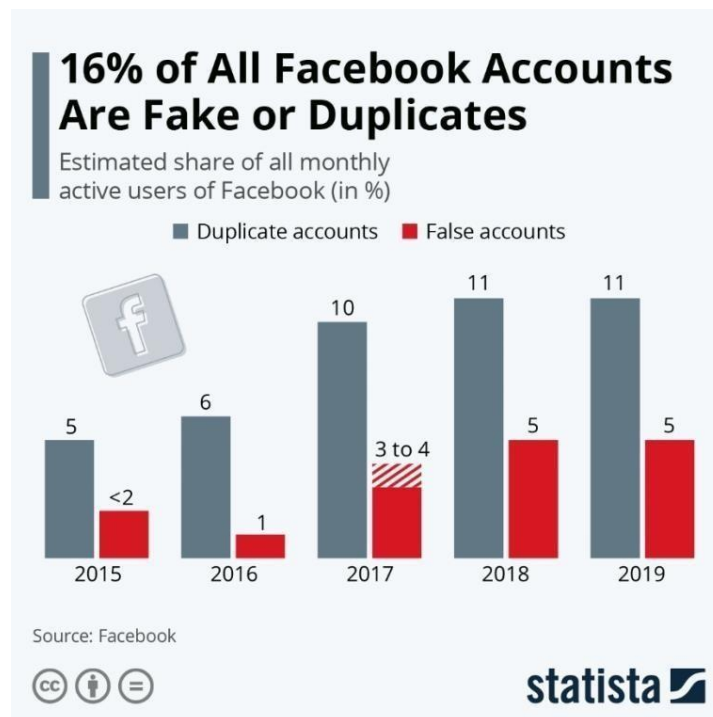


Fig1.1 : stats of increase of fake profiles as of 2019

With the fake profiles the attackers can sent the friend requests to victim and trap them, they can post false messages to damage the reputation of the targeted person and request for money/gifts ,show false romantic interests for money or try to sell false goods/items and makes the friend and family of victim vulnerable to cyber-attacks like phishing, fake links, malware attackers etc.,

1.2 OBJECTIVES:

By using an artificial neural network, we determine if the account details are true or fake. The algorithm will be taught with all false and genuine account data from previous users, after that every time we feed new test data, this algorithm training model will be applied on new test source to determine if the new account details are from genuine or false user.

Networked social media like facebook contain detailed information about users and some crime oriented users will hack into the facebook network database to steal or breach user information. To protect user data and we use Artificial neural network algorithm.

CHAPTER 2

LITERATURE SURVEY

1. Existing system:

In general, Malicious clients make counterfeit profiles to take login data from clueless clients. A phony profile will send companion solicitations to numerous clients with public profiles. when the client acknowledges the request, the proprietor of misleading profile will spam companion solicitations to anybody this client is a companion. The phony profile's items normally have joined that lead to an outside site where the harm occurs. An ignorant inquisitive client tapping the awful connection will harm their PC.

Different phony record acknowledgment strategies rely upon the examination of individual relational association profiles, with the purpose in distinctive the characteristics or a mix there of that assistance in perceiving the genuine and the phony records. Specifically, different elements are extricated from the profiles and posts, and after that Machine learning calculations are utilized in order to develop a classifier prepared for perceiving counterfeit records. For example, depicts perceiving and portraying apparition profiles in internet based social gaming applications. The article investigations a Facebook application, the web-based game "Warriors club", known to give motivations and gaming benefit to those clients who welcome their friends into the game. The creators fight that by giving such stimuli the game inspires its players to make counterfeit profiles. By introducing those phony profiles into the game, the client would expand a persuading power of a motivator for him/herself.

2. Disadvantages of the existing system:

- It only takes one fake profile to damage the computers of many.
- catching a virus to as bad as installing a rootkit turning the computer in to azombie.
- loss of personal data.
- Used for bullying the victims
- Spread fake and false information
- Damage reputation of victim
- Send false friend requests
- Lure children and teenagers with fake identity
- Makes the friends and family of victim vulnerable to cyber-attacks like phishing, fake links, malware attacks etc.,
- Financial fraud may also happen by requesting money using messenger/messageservices



Fig 2.1: a person creating a fake profile

CHAPTER 3 METHODOLOGY

3.1 Proposed system:

In this solution, we use machine learning, specifically , an artificial neural network to identify fake profiles in online social networks. we use Microsoft Excel to store old and new tamper data profiles. The data collection will be divided into training set and test set.

For the training set, the features we use to identify fake profiles are: account age, gender, user age, link in description, number of messages sent, number of requests friends sent, location entered, location by IP, etc. ..each of these parameters is checked and assigned a value.

To train ANN algorithm we are using below details from social networks

Account Age, Gender, User Age, Link_Desc, Status_Count, Friend_Count, Location, Location_IP, Status

The main purpose of all fake users is to send friend requests to normal users to hack their machine or steal their data and they will never receive many messages or have many friends following them. followers and the age of their accounts will also be less. By analysing these features, Facebook will determine if a user's profile is fake or genuine. This Facebook profile data we downloaded from Facebook website and used to train the ANN model. Here are some values from the profile dataset.

Account_Age, Gender, User_Age, Link_Desc, Status_Count, Friend_Count, Location, Location_IP, Status

10, 1, 22, 0, 1073, 237, 0, 0, 0

10, 0, 33, 0, 127, 152, 0, 0, 0

10, 1, 46, 0, 1601, 405, 0, 0, 0

10, 0, 25, 0, 704, 380, 0, 0, 0

7, 1, 34, 1, 64, 721, 1, 1, 1

7, 1, 30, 1, 69, 587, 1, 1, 1

7, 1, 36, 1, 61, 782, 1, 1, 1

7, 1, 52, 1, 96, 827, 1, 1, 1

In above dataset all strong names are the dataset segment names and all number qualities are the dataset values. As ANN won't take string esteem so we convert orientation values to 0 or 1, if male worth is 1 and assuming female worth is 0. In above dataset last segment give us data

of phony or veritable record in the event that last section contains esteem 0, account is certified in any case counterfeit. All phony record will have less number of posts as their primary expectation is to send companion demands not posts, so by investigating this elements Facebook mark that record with esteem 1 which implies it's a phony record. We are utilizing above dataset to prepare ANN model and this dataset saved inside code 'dataset' folder. After building train model we input test data with account details and ANN will give result as fake or genuine.

2. Advantages:

- Storing information on the entire network.
- Ability to work with incomplete knowledge.
- Having a distributed memory.
- Ability to make machine learning.
- Parallel processing capability.

CHAPTER 4 DESIGN

1. Software Requirements specifications:

In this project, the primary motive is to propose solution for finding a profile fake or not so in this mini project a prototype has been developed to show how the user interface works in real-time. To develop this prototype the following SRSs are used.

1. Software Requirements:

- operating system : windows 7 ultimate
- coding language : python
- Front-End : python
- Designing : Html,css,javascript
- Database : My SQL

Hardware Requirements:

- system : Pentium iv 3.5GHZ
- Hard disk : 40gb

- Monitor : 14' colormonitor
- Mouse : optical mouse
- RAM : 2GB

4.2 UML Diagrams:

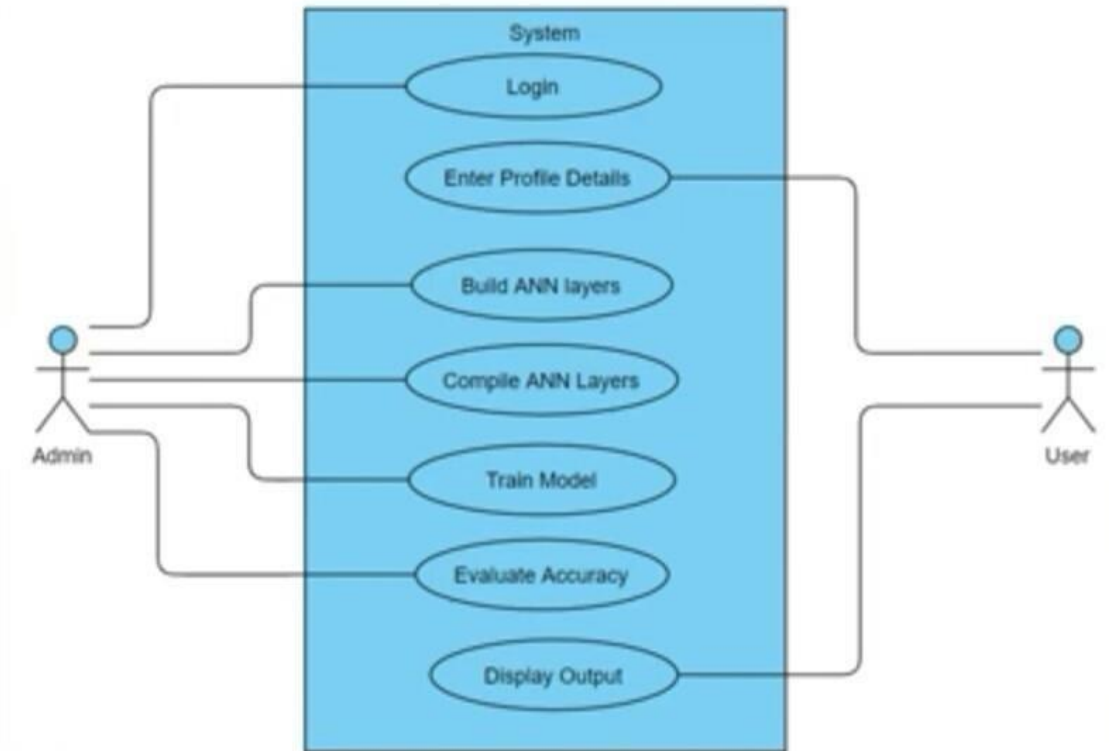


Fig 4.1 Design phase usecase diagram

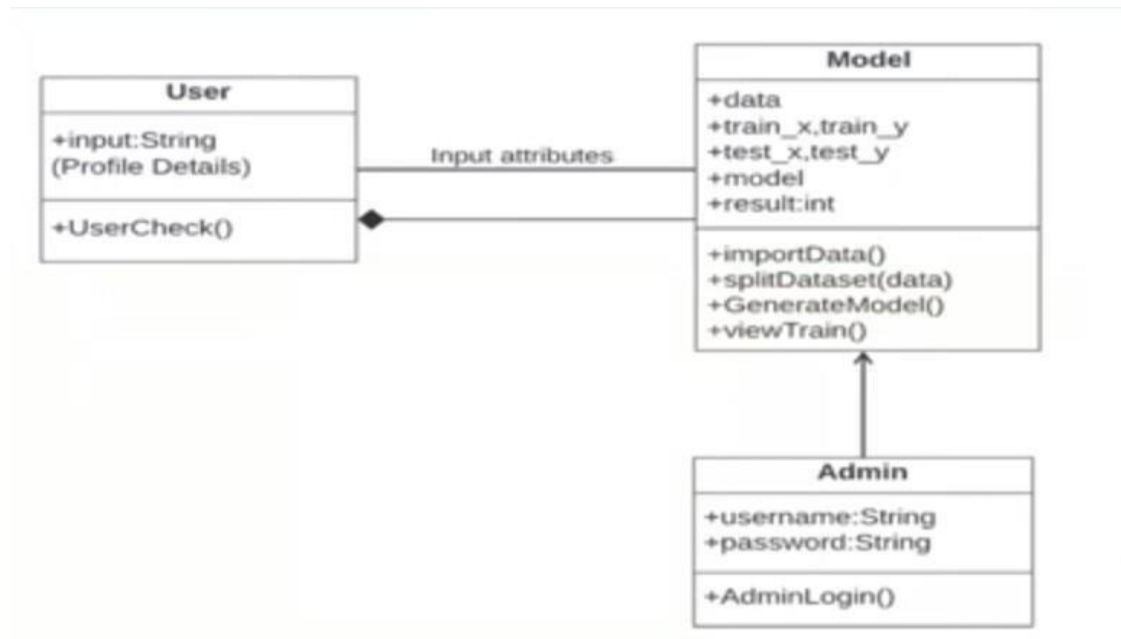


Fig 4.2:class diagram

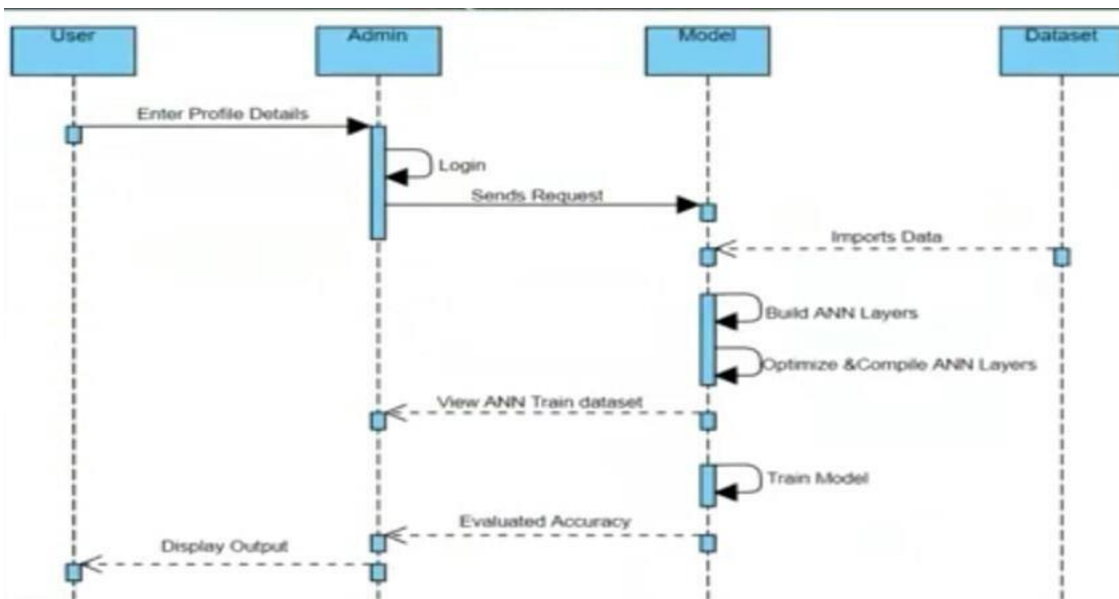


Fig4.3: sequence diagram

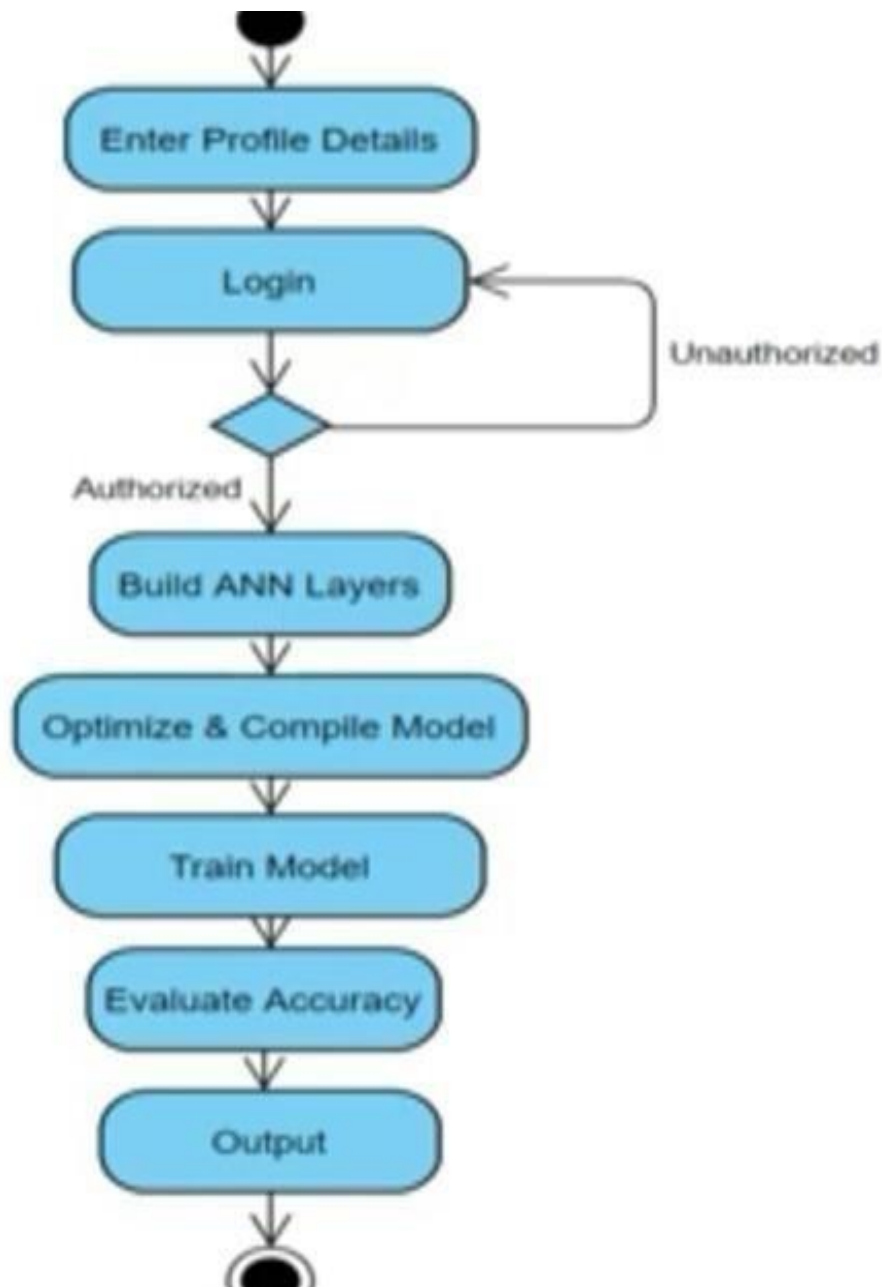


Fig4.4 Activity diagram

CHAPTER 5 IMPLEMENTATION

5.1 ANN algorithms Details:

To demonstrate how to build an image classifier based on an ANN neural network, we will build a 6-layer neural network that will identify and separate one image from another. This network that we are going to build is a very small network that we can also run on CPU. Traditional neural networks are very efficient at performing image classification with more parameters and are very time consuming if trained on a normal CPU. However, our goal is to show how to build a real-world cumulative neural network using TENSORFLOW.

If you stack neurons in a single line, it's called a layer, which is the next building block of neural networks. See below image with layers

Neural Networks are mathematical models that are used to tackle optimization problems. Neurons, the basic computing unit of neural networks, make them up. A neuron receives an input (say x), does certain calculations on it (say, multiplying it by a variable w and adding another variable b), and outputs a result (say, $z = wx + b$). To create the final output (activation) of a neuron, this value is transferred to a non-linear function called activation function (f). Activation functions come in a variety of forms. Sigmoid is a well-known activation function. A sigmoid neuron is a neuron that employs the sigmoid function as an activation function. Neurons are called according to their activation functions, and there are many different types, such as RELU and TanH.

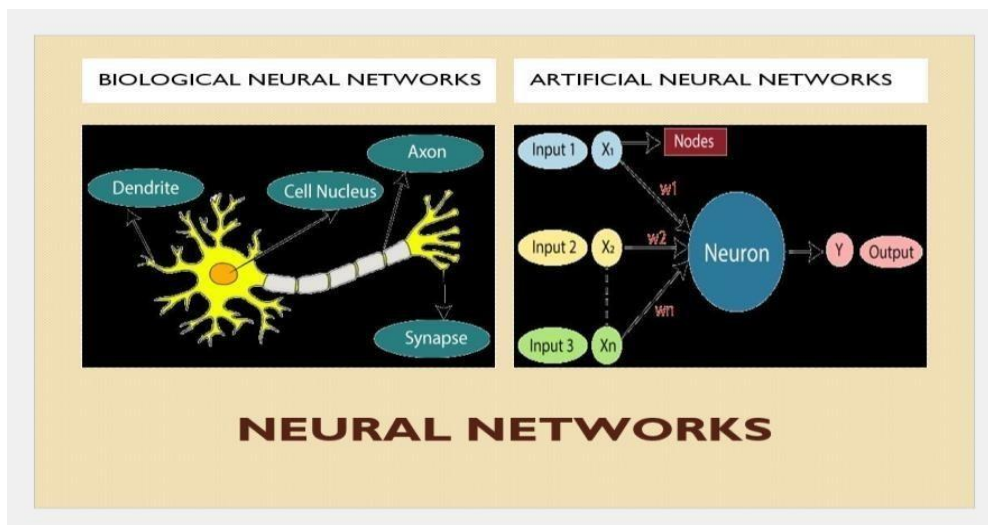


Fig5.1: similarities between artificial neural networks and biological neural networks

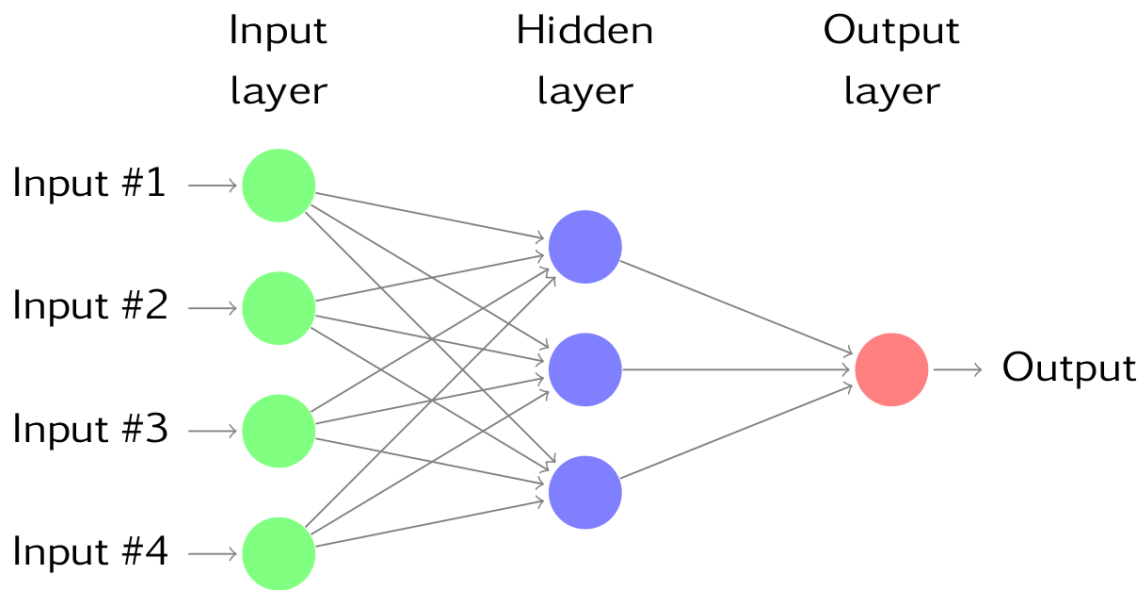


Fig5.2:architecture of neural networks

To predict image class multiple layers operate on each other to get best match layer and this process continues till no more improvement left.

Module Details:

Admin Module: Admin will login to application by using username as ‘admin’ and password as ‘admin’ and then perform below actions.

- a) a) Create an ANN Train Model: To create a train model, the administrator will submit the profile dataset to the ANN algorithm. Using new account test data, this train model can predict if an account is phoney or real.
- b) View ANN Train Dataset: Using this module admin can view all dataset used to trainANN model.

User Module: This application allows any user to submit test data for a new account and run the ANN algorithm. The ANN algorithm will take new test data and apply a train model to determine if the data is phoney or real.

5.2 Data collection :

For implementing the application, sample dataset is collected from the Kaggle website

And the dataset will be like in terms of **Account_Age, Gender, User_Age, Link_Desc, Status_Count, Friend_Count, Location, Location_IP**, Status and example of dataset is

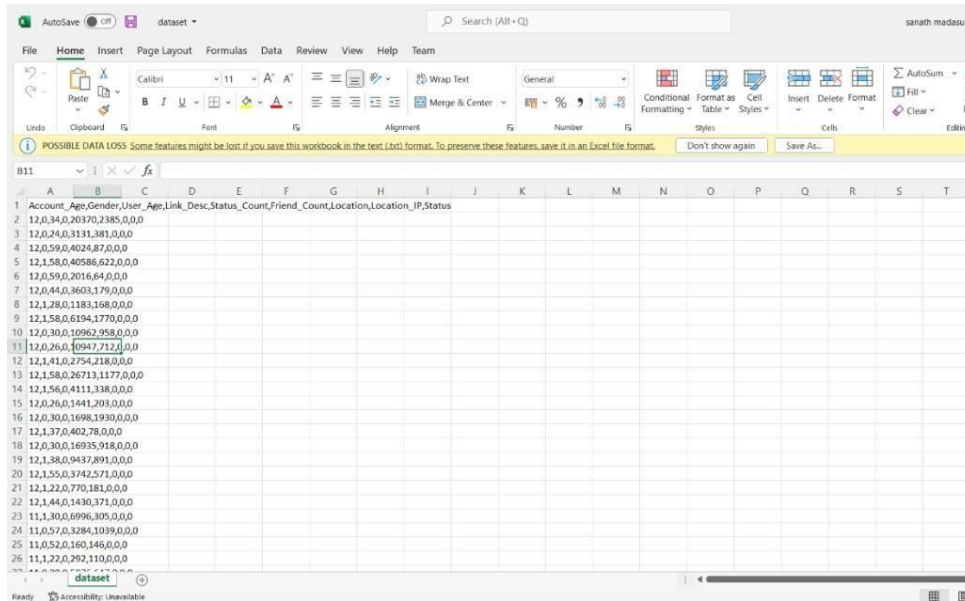


Fig picture of facebook dataset

5.3 Source code:

Deploy this application on DJANGO server and then run in browser enter URL as <http://localhost:8000/index.html> to get the application window

5.3.1 Views.py

```
views - Notepad
File Edit View

username = request.POST.get('username', False)
password = request.POST.get('password', False)
if username == 'admin' and password == 'admin':
    context= {'data':'welcome'+username}
    return render(request, 'AdminScreen.html', context)
else:
    context= {'data':'login failed'}
    return render(request, 'Admin.html', context)

def importdata():
    balance_data = pd.read_csv('C:/Users/sanat/Desktop/miniproject/Use of Artificial Neural Networks to Identify FakeProfiles/Profile/dataset/dataset.txt')
    balance_data = balance_data.abc()
    rows = balance_data.shape[0] # gives number of row count
    cols = balance_data.shape[1] # gives number of col count
    return balance_data

def splitdataset(balance_data):
    X = balance_data.values[:, 0:8]
    y = balance_data.values[:, 8]
    y_ = y.reshape(-1, 1)
    encoder = OneHotEncoder(sparse=False)
    Y = encoder.fit_transform(y_)
    print(Y)
    train_x, test_x, train_y, test_y = train_test_split(X, Y, test_size=0.2)
    return train_x, test_x, train_y, test_y

def UserCheck(request):
    if request.method == 'POST':
        data = request.POST.get('l', False)
        input = 'Account_Age,Gender,User_Age,Link_Desc,Status_Count,Friend_Count,Location,Location_IP,IP'
        input += data + "\n"
        f = open('C:/Users/sanat/Desktop/miniproject/Use of Artificial Neural Networks to Identify FakeProfiles/Profile/dataset/test.txt', 'w')
```

```
views - Notepad
File Edit View

input+=data+"\n"
f = open("C:/Users/sanat/Desktop/miniproject/Use of Artificial Neural Networks to Identify FakeProfiles/Profile/dataset/test.txt", "w")
f.write(input)
f.close()
test = pd.read_csv("C:/Users/sanat/Desktop/miniproject/Use of Artificial Neural Networks to Identify FakeProfiles/Profile/dataset/test.txt")
test = test.values[:, 0:8]
predict = model.predict_classes(test)
print(predict[0])
msg = ""
if str(predict[0]) == '0':
    msg = "Given Account Details Predicted As Genuine"
if str(predict[0]) == '1':
    msg = "Given Account Details Predicted As Fake"
context= {'data':msg}
return render(request, 'User.html', context)

def GenerateModel(request):
    global model
    data = importdata()
    train_x, test_x, train_y, test_y = splitdataset(data)
    model = Sequential()
    model.add(Dense(200, input_shape=(8,), activation='relu', name='fc1'))
    model.add(Dense(200, activation='relu', name='fc2'))
    model.add(Dense(2, activation='softmax', name='output'))
    optimizer = Adam(lr=0.001)
    model.compile(optimizer, loss='categorical_crossentropy', metrics=[accuracy])
    print("ANN Neural Network Model Summary:")
    print(model.summary())
    model.fit(train_x, train_y, verbose=2, batch_size=5, epochs=200)
    results = model.evaluate(test_x, test_y)
    ann_acc = results[1] * 100
    context= {'data':ANN Accuracy: '+str(ann_acc)}
    return render(request, 'AdminScreen.html', context)

Ln 1, Col 1
36°C Clear 100% Windows (CRLF) UTF-8 08:40 PM 14-05-2022
```

```
views - Notepad
File Edit View

def GenerateModel(request):
    global model
    data = importdata()
    train_x, test_x, train_y, test_y = splitdataset(data)
    model = Sequential()
    model.add(Dense(200, input_shape=(8,), activation='relu', name='fc1'))
    model.add(Dense(200, activation='relu', name='fc2'))
    model.add(Dense(2, activation='softmax', name='output'))
    optimizer = Adam(lr=0.001)
    model.compile(optimizer, loss='categorical_crossentropy', metrics=[accuracy])
    print("ANN Neural Network Model Summary:")
    print(model.summary())
    model.fit(train_x, train_y, verbose=2, batch_size=5, epochs=200)
    results = model.evaluate(test_x, test_y)
    ann_acc = results[1] * 100
    context= {'data':ANN Accuracy: '+str(ann_acc)}
    return render(request, 'AdminScreen.html', context)

def ViewTrain(request):
    if request.method == 'GET':
        strdata = '<table border=1 align=center width=100%><tr><th><font size=4 color=white>Account Age</th><th><font size=4 color=white>Gender</th><th><font size=4 color=white>User Age</th><th><font size=4 color=white>Link Description</th><th><font size=4 color=white>Status Count</th><th><font size=4 color=white>Friend Count</th><th><font size=4 color=white>Location</th><th><font size=4 color=white>Location IP</th><th><font size=4 color=white>Profile Status</th></tr><tr>'
        data = pd.read_csv("C:/Users/sanat/Desktop/miniproject/Use of Artificial Neural Networks to Identify FakeProfiles/Profile/dataset/dataset.txt")
        rows = data.shape[0] # gives number of row count
        cols = data.shape[1] # gives number of col count
        for i in range(rows):
            for j in range(cols):
                strdata+='%<td><font size=3 color=white>'+str(data.iloc[i,j])+'</font></td>'
            strdata+='%<tr><tr>'
        context= {'data':strdata}
        return render(request, 'ViewData.html', context)

Ln 1, Col 1
36°C Clear 100% Windows (CRLF) UTF-8 08:40 PM 14-05-2022
```

```
views - Notepad
File Edit View

from django.shortcuts import render
from django.template import RequestContext
from django.contrib import messages
from django.http import HttpResponseRedirect
import pandas as pd
from sklearn.model_selection import train_test_split
from keras.models import Sequential
from keras.layers.core import Dense, Activation, Dropout
from keras.callbacks import EarlyStopping
from sklearn.preprocessing import OneHotEncoder
from keras.optimizers import Adam

global model

def index(request):
    if request.method == 'GET':
        return render(request, 'index.html', {})

def User(request):
    if request.method == 'GET':
        return render(request, 'User.html', {})

def Admin(request):
    if request.method == 'GET':
        return render(request, 'Admin.html', {})

def AdminLogin(request):
    if request.method == 'POST':
        username = request.POST.get('username', False)
        password = request.POST.get('password', False)
        if username == 'admin' and password == 'admin':
            return HttpResponseRedirect('/admin/')
        else:
            messages.error(request, 'Invalid username or password')
            return HttpResponseRedirect('/adminlogin/')

Ln 1, Col 1
30°C Clear
Windows (CRLF) UTF-8
08:39 PM 14-05-2022
```

5.3.2 Admin Module :

```
Admin - Notepad
File Edit View

{% load static %}
<html>
<head>
<title> Fake Profiles Detection</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="{% static 'default.css' %}" rel="stylesheet" type="text/css" media="screen" />
<script LANGUAGE="JavaScript">
function validate() {
    var x=document.forms["f1"]["username"].value;
    var y=document.forms["f1"]["password"].value;
    if(x==null || x=="") {
        window.alert("Username must be enter");
        document.f1.t1.focus();
        return false;
    }
    if(y==null || y=="") {
        window.alert("Password must be enter");
        document.f1.t2.focus();
        return false;
    }
    return true;
}
</script>
</head>
<body>
<div id="wrapper">
<div id="header">
<div id="logo">
<h1><font color="orange" size=6>Use of Artificial Neural Networks to Identify Fake
Profiles</font></h1>
Ln 36, Col 3
30°C Clear
Windows (CRLF) UTF-8
08:56 PM 14-05-2022
```

```

Admin - Notepad
File Edit View

<div id="menu">
<ul>
<li><a href="{% url 'index' %}">Home</a></li>
<li><a href="{% url 'Admin' %}">Admin</a></li>
<li><a href="{% url 'User' %}">Users</a></li>
</ul>
</div>
<div class="entry">
<br><br><br>
<font size="" color="white"><center>{{ data }}</center></font>
<br>
<center><font size="3" color="white">Admin Login Screen</font></center>
<form name="f1" method="post" action="{% url 'AdminLogin' %}" OnSubmit="return validate()"
{% csrf_token %}
<br><br>
<table align="center" width="35%" class="notepad">
<tr><th align="left">Username
<td>&nbsp;&nbsp;&nbsp;&nbsp;<input type="text" name="username" value="" class="form-control">
<div id="nameid"></div>
</td>
</tr>

<tr><th align="left">Password
<td>&nbsp;&nbsp;&nbsp;&nbsp;<input type="password" name="password" value="" class="form-control">
</td>
</tr>
<tr>
<td></td>
<td>
<input type="submit" value="login">
</td>
</tr>
</table>

```

5.3.3 User Module:

```

User - Notepad
File Edit View

{% load static %}
<html>
<head>
<title>Fake Profiles Detection</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="{% static 'default.css' %}" rel="stylesheet" type="text/css" media="screen" />
<script LANGUAGE="JavaScript">
function validate()
{
var x=document.forms["f1"]["t1"].value;
if(x==null || x=="")
{
window.alert("Please enter account details");
document.f1.t1.focus();
return false;
}

return true;
}
</script>
</head>
<body>
<div id="wrapper">
<div id="header">
<div id="logo">
<h1><font color="orange" size=6>Use of Artificial Neural Networks to Identify Fake
Profiles</font></h1>
<marquee><font color="pink" size=4>Fake Profiles Detection</font></marquee>
</div>

</div>

```

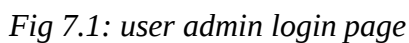





Fig 7.2: welcome page after successful login

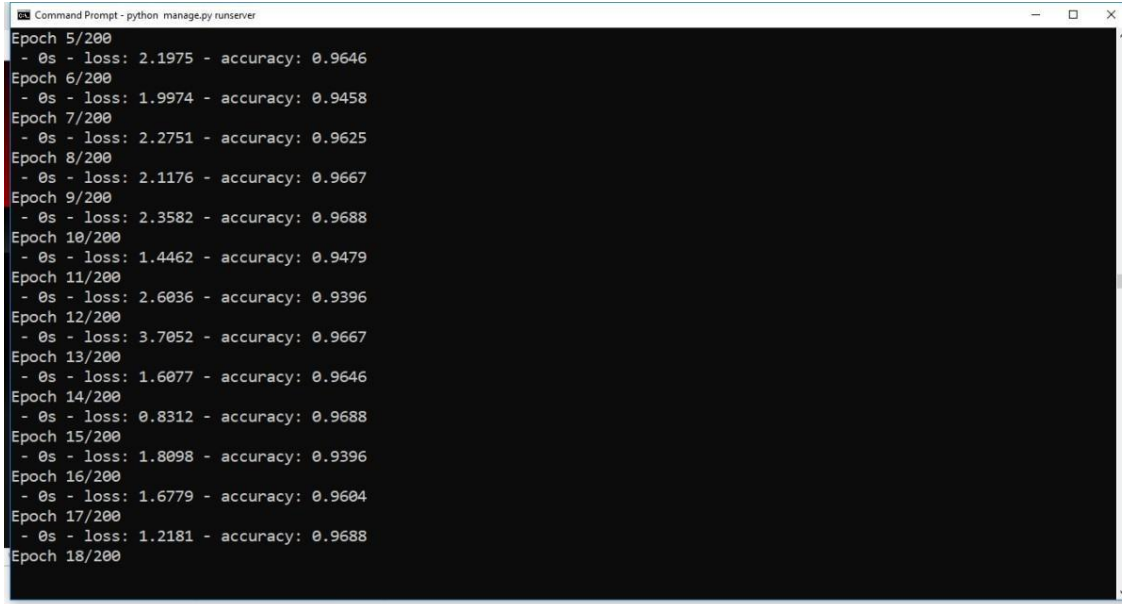
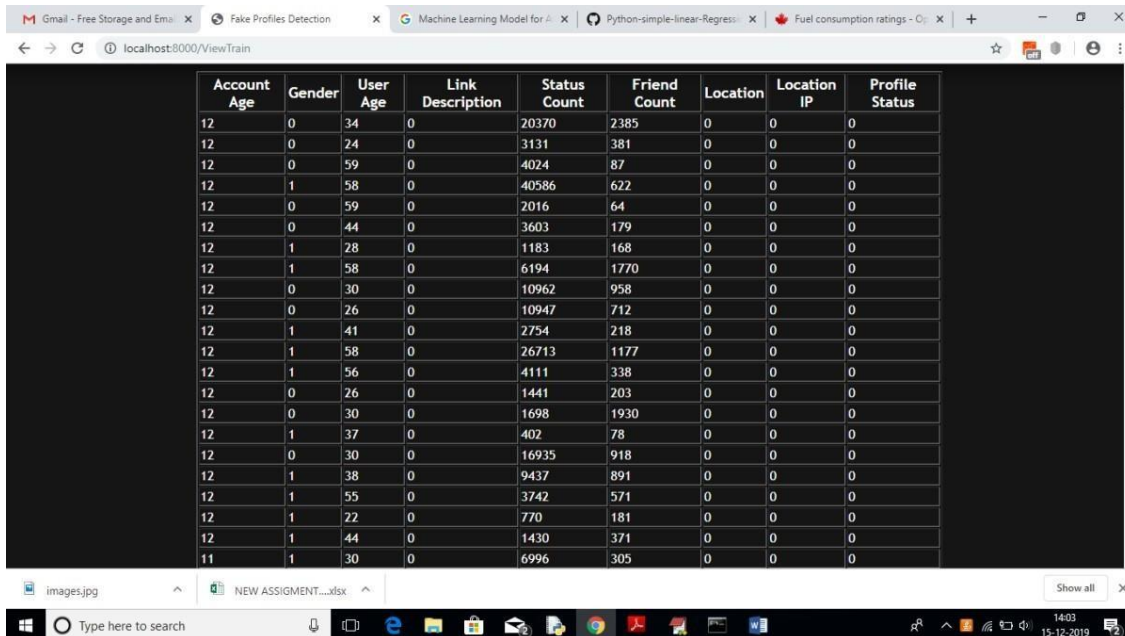


fig7.3: training the database using ANN algorithm



Fig7.4: figure showing accuracy of trained dataset



Account Age	Gender	User Age	Link Description	Status Count	Friend Count	Location	Location IP	Profile Status
12	0	34	0	20370	2385	0	0	0
12	0	24	0	3131	381	0	0	0
12	0	59	0	4024	87	0	0	0
12	1	58	0	40586	622	0	0	0
12	0	59	0	2016	64	0	0	0
12	0	44	0	3603	179	0	0	0
12	1	28	0	1183	168	0	0	0
12	1	58	0	6194	1770	0	0	0
12	0	30	0	10962	958	0	0	0
12	0	26	0	10947	712	0	0	0
12	1	41	0	2754	218	0	0	0
12	1	58	0	26713	1177	0	0	0
12	1	56	0	4111	338	0	0	0
12	0	26	0	1441	203	0	0	0
12	0	30	0	1698	1930	0	0	0
12	1	37	0	402	78	0	0	0
12	0	30	0	16935	918	0	0	0
12	1	38	0	9437	891	0	0	0
12	1	55	0	3742	571	0	0	0
12	1	22	0	770	181	0	0	0
12	1	44	0	1430	371	0	0	0
11	1	30	0	6996	305	0	0	0

fig7.5 : Trained dataset

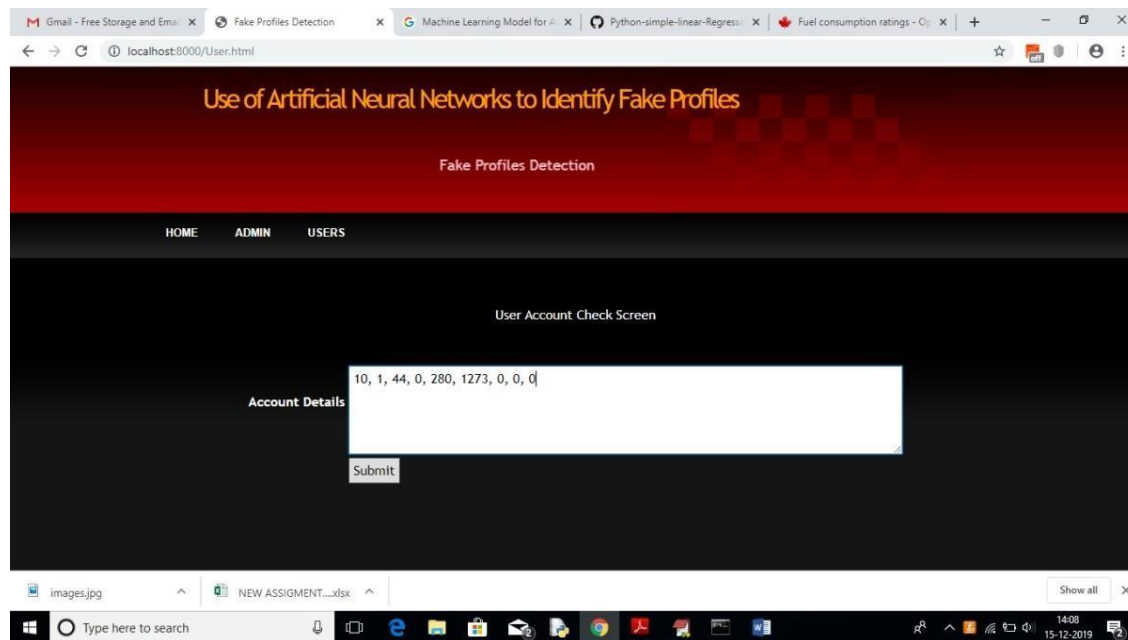


Fig7.6: after logging into user profile we have to submit profile

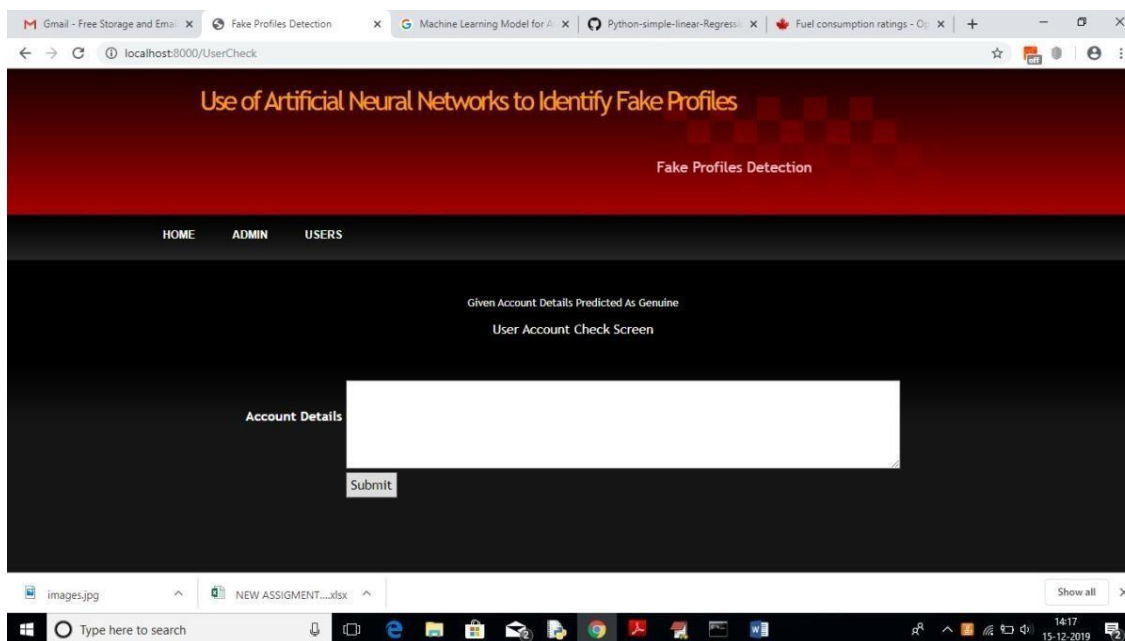


fig7.7: showing the given profile is genuine

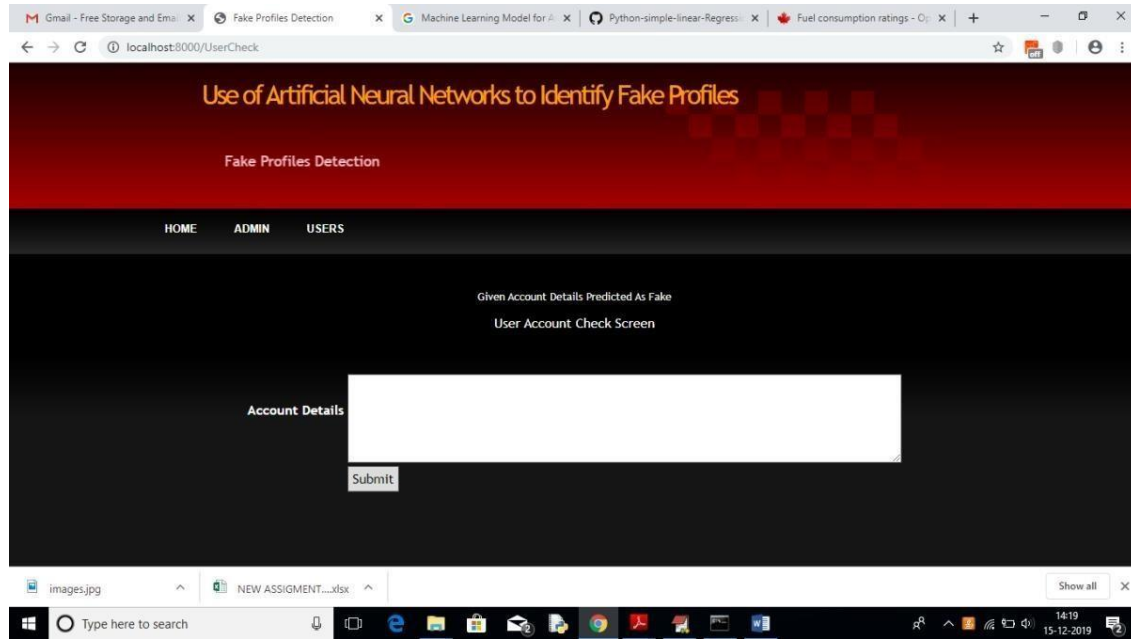


Fig7.8: showing the given profile is fake

CHAPTER 7 CONCLUSION AND FUTURE SCOPE

In this task, we proposed AI calculation to be specific counterfeit brain organization to recognize the phony profiles from the informal community destinations.

we took the Facebook dataset to distinguish the phony profiles. This would permit the introduced profound learning calculation to gain proficiency with the examples of bot conduct by back propagation, minimizing the last expense work and changing every neuron's weight and bias. And likewise with this model application we can distinguish the profile which is genuine or not so the clients can save them from the aggressors we frame the classes and libraries included. We additionally talk about the sigmoid capacity and how are not entirely settled and utilized. We likewise consider the boundaries of the interpersonal organization page which are the most vital to our answer.

The future extent of this task is exceptionally high as this application can be Used to recognize the phony profiles by every client so they can be utilizing the Facebook with guarantee and the long-range informal communication

locales ought to carry out this calculation in fabricated so the need of the application and time taken to confirm the profiles will be diminished.

CHAPTER 8 BIBLIOGRAPHY

1FrancescoBuccafurri, Gianluca Lax,Denis Migdal, Serena Nicolazzo, Antonino Nocera and Christophe Rosenberger"Contrasting False Identities in Social Networks by Trust Chains and Biometric Reinforcement " International Conference on Cyberworlds vol 5,2017.

2MohamedTorky, Ali Meligy and Hani Ibrahim"Recognizing Fake Identities In Online Social Networks Based on a Finite Automaton Approach"International Journal of ComputerApplications, 2016.

3SuprajaGurajala, Joshua S White, Brian Hudson, Brian R Voter, and Jeanna N Matthews"Profile characteristics of fake Twitter accounts"Big Data &Society, July–December 2016: 1–13.

4Simran Jit. Kaur. Tuteja, “A survey on classification algorithms for email spam filtering,” International Journal Eng. Sci., vol. 6, no. 5, pp. 5937–5940, 2016.

5M.A. Devmane and N.K. Rana "Detection and Prevention of Profile Cloning in Online Social Networks"IEEE International Conference on Recent Advances and Innovations inEngineering,May 09-11, 2014.

6SaraKeretna ,Ahmad Hossny and Doug Creighton "Recognising User Identity in Twitter Social Networks via Text Mining"IEEE International Conference on Systems, Man, and Cybernetics,2013.

<https://www.statista.com/topics/1164/social-networks/>

7 <https://www.cnbc.com/2018/01/31/facebook-earnings-q4-2017- arpu.html>

8 <https://www.cnet.com/news/facebook-breach-affected-50-millionpeople/>

9 <https://www.facebook.com/policy.php>

10Qiang Cao, Michael Sirivianos, Xiaowei Yang, and Tiago Pregueiro. 2012. Aiding the detection of fake accounts in large scale social online services. In Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation (NSDI'12).USENIX Association, Berkeley, CA, USA, 15-



