Use of Artificial Neural Networks Identify Fake Profiles

A PROJECT REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY,ANANTAPUR

Submitted in fulfillment of the Requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

N.Udayasri   19F71A0575

Under the guidance of

Mr.D.Eswaraiah, M.Tech.,

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRI SAI INSTITUTE OF TECHNOLOGY AND SCIENCE

(Affiliated to JNTUA, Ananthapuramu)

Rayachoty-516269, Annamayya (Dist.) – AP

2019-2023

SRI SAI INSTITUTE OF TECHNOLOGY AND SCIENCE
(Affiliated to JNTUA, Ananthapuramu)
Rayachoty-516 270, Annamayya (Dist.) – AP
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project entitled "Use of Artificial Neural Networks to Identify Fake Profiles" is being submitted by

N.Udayasri   (19F71A0575)

In partial fulfillment for the award of the Degree of Bachelor of Technology in Computer science & engineering to the Sri Sai Institute of Technology and Science, Rayachoty ,is a record of benefited work carried out under my guidance and supervision. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Project Guide   **Head of the Department**

Mr.D.ESWARAIAH,M.Tech.,   **Mr.P.SUBHAN BASHA**,M.Tech.,

Assistant Professor,   Assistant Professor and HOD

Department of CSE,   Department of CSE,

SSITS, Rayachoty-516269   SSITS, Rayachoty-516269

Submitted for External Viva –Voce Examination held on

Guide   **Head of the Department**

Internal Examiner   **External Examiner**

<u>Acknowledgement</u>

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that I have now the opportunity to express my guidance for all of them.

We deeply thank our chief patron Sri.K.P.V.SUBBAIAH Garu, Sri Sai Institute of technology and Science, with whose support and provision this project has been carried out with required infrastructure.

We wish to thank our Assistant Director, Dr.M.SUDHAKAR REDDY, Ph.D. His motivation words that created burning desire in us and that eventually led to successfully completing this project work.

The man who has helped us a lot in times of trouble, and who helped us in recovering from intricate problems is Dr.V.BALAJI,Ph.D. principal, SSITS College of Engineering, Rayachoty. I am thankful to him.

Foremost, we gratefully acknowledge my sincere gratitude to our respected Mr.P.SUBHAN BASHA,M.Tech., Assistant Professor, Head of the Department, Department of Computer Science and Engineering, SSITS College of Engineering, Rayachoty for her valuable guidance, timely suggestions, and lively intense throughout the work. We always thankful to her for providing me the opportunity to do work.

The real mentor and motivator of this project Mr.D.ESWARAIAH,M.Tech.,Assistant Professor of Computer Science and Engineering Department, SSITS College of Engineering,Rayachoty. His wide knowledge and logical way of thinking have made a deep impression on me. His understanding, encouragement and personal guidance have provided the basis for this thesis. His source of inspiration for innovative ideas and his kind support is well to all his students and colleagues.

Last but far from least, We also thank my family members, my friends, staff and faculty members of CSE Dept for their moral support and constant encouragement, We are very much thankful to one and all that helped our successful completion of the project.

<div align="right">

With gratitude

N.Udayasri   19F71A0575

V.Praveena   19F71A0548

P.Sravani   19F71A0570

M.Supriya   19F71A0572

K.Sai Sindura 19F71A0584

</div>

CONTENTS

<u>ABSTRACT</u>

In this project, we use machine learning, namely an artificial neural network to determine what the chances that Facebook friend request is authentic are or not. We also outline the classes and libraries involved. Furthermore, we discuss the sigmoid function and how the weights are determined and used. Finally, we consider the parameters of the social network page which are utmost important in the provided solution.

The other dangers of personal data being obtained for fraudulent purposes are the presence of bots and fake profiles. Bots are programs that can gather information about the user without the user even knowing. This process is known as web scraping. What is worse is that this action is legal. Bots can be hidden or come in the form of a fake friend request on a social network site to gain access to private information

ACRONYMS & ABBREVIATIONS

i. **HTML:**   Hyper Text Markup Language.

ii. **XML:**   Extensible Markup Language.

iii. **IDE:**   Integrated Development Environment

iv. **PHP:**   Hyper Text Preprocessor

v. **RDBMS:**   Relational Database Management System.

vi. **GUI:**   Graphical User Interface

vii. **HTTP:**   Hyper Text Transfer Protocol

viii. **API:**   Application Programming Interface

ix. **E-R:**   Entity-Relationship

x. **UML:**   Unified Modeling Language

xi. **OOAD:**   Object-Oriented Analysis & Design.

LIST OF FIGURES

<div align="center">

CHAPTER – 1
INTRODUCTION

</div>

The chapter gives brief introduction of the project.

1.1 PROJECT INTRODUCTION

In 2017 Facebook reached a total population of 2.46 billion users making it the most popular choice of social media. Social media networks make revenues from the data provided by users. The average user does not know that their rights are given up the moment they use the social media network's service. Social media companies have a lot to gain at the expense of the user. Every time a user shares a new location, new photos, likes, dislikes, and tag other users in content posted, Facebook makes revenue via advertisements and data. More specifically, the average American user generates about $26.76 per quarter. That number adds up quickly when millions of users are involved. In today's digital age, the ever-increasing dependency on computer technology has left the average citizen vulnerable to crimes such as data breaches and possible identity theft. These attacks can occur without notice and often without notification to the victims of a data breach. At this time, there is little incentive for social networks to improve their data security. These breaches often target social media networks such as Facebook and Twitter. They can also target banks and other financial institutions. There seems to be a newsworthy issue involving social media networks getting hacked every day. Recently, Facebook had a data breach which affected about 50 million users. Facebook provides a set of clearly defined provisions that explain what they do with the user's data. The policy does very little to prevent the constant exploitation of security and privacy. Fake profiles seem to slip through Facebook's built-in security features. The other dangers of personal data being obtained for fraudulent purposes are the presence of bots and fake profiles. Bots are programs that can gather information about the user without the user even knowing. This process is known as web scraping. What is worse, is that this action is legal. Bots can be hidden or come in the form of a fake friend request on a social network site to gain access to private information. The solution presented in this paper intends to focus on the dangers of a bot in the form of a fake profile on your social media. This solution would come in the form of an algorithm. The language that we chose to use is Python. The algorithm would be able to determine if a current friend request that a

user gets online is an actual person or if it is a bot or it is a fake friend request fishing for information. Our algorithm would work with the help of the social media companies, as we would need a training dataset from them to train our model and later verify if the profiles are fake or not. The algorithm could even work as a traditional layer on the user's web browser as a browser plug-in.

Sybil rank was designed in late 2012, to efficiently identify fake profiles through a ranking graph-based system. The algorithm uses a seed selection method combined with early terminated random walks to propagate trust. Its computational cost is measured in O (nlogn). Profiles are ranked according to the number of interactions, tags, wall posts, and friends over time. Profiles that have a high rank are considered to be real with fake profiles having a low rank in the system. Unfortunately, this technique was found to be mostly unreliable because it failed to take into account the possibility that real profiles can be ranked low and fake profiles can be ranked high. Strode and Mishra proposed a different approach which is a sequence of steps to detect fake profiles. They used the Facebook graph API tool to gain access to numerous profiles and wrote a script to extract the viewed information. Later on, this extracted information forms the attributes the classifier will use in their algorithm. First, the data is in JSON format, which is further parsed to a structured format (CSV) that is easier readable by machine learning techniques. These comma separated values will later make the classifier more efficient. The authors tried unsupervised and also supervised machine learning techniques. In this case, supervised machine learning techniques had a higher accuracy rate of almost 98%. For supervised machine learning, they split up the dataset into training and testing sets. They used 80% of the samples to train the classifier and the rest to test it. After the algorithm runs, there is feedback provided to the profile, requiring it to submit identification to prove it is not a fake profile. Profiles are processed on mass to extract features. Resilient Back Propagation algorithm in neural networks algorithm combined with support vector machines is used in the classification of fake profiles. Sybil Frame uses multi-stage level classification. Approaches include content-based and structure based. Content-based approach explores the dataset and extracts information used to calculate prior information about nodes and edges. Structure-based approach correlates nodes using Markov random field and loopy belief propagation which employs previous information. The content-based approach is used in the first stage of Sybil Frame and

Structure-based approach is used in the second stage of Sybil Frame technique. Clickstreams are analyzed, and Friend recommendations are examined in stage I. Vote Trust uses a voting based system that pulls user activities to find fake profiles using trust-based vote assignment and global votes total. It is considered as the first line of defense due to limitations which include real accounts that were already compromised being sold IN "**FAKE PROFILE IDENTIFICATION USING MACHINE LEARNING**" The social network, a crucial part of our life is plagued by online impersonation and fake accounts. According to the 'Community Standards Enforcement Report' published by Facebook on March 2018, about 583 million fake accounts were taken down just in quarter 1 of 2018 and as many as 3-4% of its active accounts during this time were still fake. In this project, we propose a model that could be used to classify an account as fake or genuine. This model uses Support Vector Machine as classification technique and can process a large dataset of accounts at once, eliminating the need to evaluate each account manually. The community of concern to us here is Fake Accounts and our problem can be said to be a classification or a clustering problem. In the present generation, the social life of everyone has become associated with the online social networks. Adding new friends and keeping in contact with them and their updates has become easier. The online social networks have impact on the science, education, grassroots organizing, employment, business, etc. Researchers have been studying these online social networks to see the impact they make on the people. Teachers can reach the students easily through this making a friendly environment for the students to study, teachers nowadays are getting themselves familiar to these sites bringing online classroom pages, giving homework, making discussions, etc. which improves education a lot. The employers can use these social networking sites to employ the people who are talented and interested in the work, their background check can be done easily.IN "**FAKE ACCOUNT DETECTION USING ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS"** Social media platform is extremely popular these days and because of it's open nature it leads to increase in cybercrimes, harassments, spam messages through fake accounts so the project focuses on detecting fake accounts on social media using SVM classifier(Support Vector Machine) and Neural Network model. Using a new algorithm that is SVM-NN. Here we are using MIB dataset which is a predefined dataset that consists of a list of features available in social media like the friend-count, request accept count,

etc. Like that 16 top categories are the application for everything social media platform. Detecting which activity to perform fake account detection we can't predefine as it will be different for different accounts and social media platforms this issue is getting solved by SVM by selecting which of these activities given by the user that we have taken to identify which account is fake and which is not Web-based life has been instilled into our general public today that is unthinkable for individuals to dispose of it. Independent of ages, each age has entertained themselves into these stages that have made them dependent on it. Today, electronic life plays an irrefutably progressively basic activity since customers contribute a huge amount and they consider online being a basic wellspring of information about issues that customers are stressed about. Platforms like Facebook, YouTube, Snap chat and so on rehearses an age decent variety culture, where distinctive age individuals transfer and speak with outsiders. Online progression toward ending up being generators of culture and new examples. The movement of electronic life, like YouTube, opens a new world for enhancement of virtual relational associations connected by basic inclinations. Clients contribute continuously greater imperativeness online to structure. Electronic life turns into a important part of their life and virtual frameworks have essential impacts on enhancement of social measures and watches that sway the movement of significant great and influence life of the individual impressively. Apart from being an incredible stage to improve one's correspondence level, numerous web-based social networking stages have additionally affected numerous issues throughout a person's life. Assailants follow having OSNs customer accounts betray themselves to be someone else, they tend to use photos and profiles that they get by a veritable individual without their data and made misleadingly, they announce fake news, additionally, take singular data. These phony records are for the most part called fakers In the two cases, such phony accounts harmfully affect clients, and their intentions would be something besides well-meaning goals as they generally, flood spam messages, or take private information. They are quick to phish individual guileless clients to fake connections that lead to sex trick, human dealing, and even political astroturfing Insights show that 40% of watchmen in the United States, besides, 18% of adolescents have mind-blowing stress over the usage of fake records and bots utilizing online systems administration and affect things. YouTube permits clients to control their remark area physically through settings

include. YouTube clients need to physically change the settings that permit them to allow certain remarks. Ongoing occasions have watched many ruthless remarks in Recordings highlighting minor-matured young ladies. The remark area has been seen to be the most influenced by this issue. YouTube has likewise had issues in its proposal framework. Recordings that are prescribed while observing surely driving clients to another circle entirety. These issues were additionally seen in day by day family loggers, which included small children in their posts. It was seen that there were youngster ruthless remarks including time-stamp that were referenced trailed by unparliamentarily, injurious word.

1.2 Project Overview

Each input neuron would be a different, previously chosen feature of each profile converted into a numerical value (e.g., gender as a binary number, female 0 and male 1) and if needed, divided by an arbitrary number (e.g., age is always divided by 100) to minimize one feature having more influence on the result than the other. The neurons represent nodes. Each node would be responsible for exactly one decision-making process.

1.3 Existing System:-

Malicious users create fake profiles to phish login information from unsuspecting users. A fake profile will send friend requests to many users with public profiles. These counterfeit profiles bait unsuspecting users with pictures of people that are considered attractive. Once the user accepts the request, the owner of the phony profile will spam friend requests to anyone this user is a friend.

The fake profile's contents typically have links that lead to an external website where the damage happens. An unaware curious user clicking the bad link will damage their computer. The cost can be as simple as catching a virus to as bad as installing a rootkit turning the computer into a zombie. While Facebook has a rigorous screening to keep these fake accounts out, it only takes one fake profile to damage the computers of many.

Disadvantages:

Neural networks can be difficult to interpret.

Neural networks may over fit and not generalize well to new data.

Artificial neural networks require access to large amount of data , which can raise concerns about data privacy and security.

The development and training of a neural network can be expensive, especially for smaller organizations.

1.4 Proposed System

In our solution, we use machine learning, namely an artificial neural network to determine what the chances that a friend request is authentic are or not.

We utilize Microsoft Excel to store old and new fake data profiles. The algorithm then stores the data in a data frame. This collection of data will be divided into a training set and a testing set. We would need a data set from the social media sites to train our model.

For the training set, the features that we use to determine a fake profile are Account age, Gender, User age, Link in the description, Number of messages sent out, Number of friend requests sent out, Entered location, Location by IP, Fake or Not. Each of these parameters is tested and assigned a value. For example, for the gender parameter if the profile can be determined to be a female or male a value of (1) is assigned to the training set for Gender. The same process is applied to other parameters. We also use the country of origin as a factor

Advantages

- Artificial neural networks useful for real-time identification of fake profiles
- ANN can process large volumes of data quickly and efficiently, which makes them a useful tool for identifying fake profiles in real-time.
- Neural networks can be easily scaled up or down depending on the size of the dataset being analysed.

- After training, neural networks can be automatically detect fake profiles, reducing the need for human intervention in the analysis process.

CHAPTER 2

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- ❖ **ECONOMICAL FEASIBILITY**
- ❖ **TECHNICAL FEASIBILITY**
- ❖ **SOCIAL FEASIBILITY**

2.1 ECONOMICAL FEASIBILITY

 This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

2.2 TECHNICAL FEASIBILITY

    This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have modest requirement, as only minimal or null changes are required for implementing this system.

2.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER - 3

SOFTWARE REQUIREMENT

SPECIFICATION

Gives the details of platform specifications, Hardware, and Software specifications.

This chapter provides the details of the project's need based survey, system requirements, Hardware Requirements, Software Requirements, and System Requirements.

3.1 Functional Requirements:-

In software engineering, a functional requirement defines a system or its component. It describes the functions software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional software requirements help you to capture the intended behavior of the system. This behavior may be expressed as functions, services or tasks or which system is required to perform.

3.2 Non-Functional Requirements:-

A non-functional requirement defines the quality attribute of a software system. They represent a set of standards used to judge the specific operation of a system. Example, how fast does the website load? A non-functional requirement is essential to ensure the usability and effectiveness of the entire software system. Failing to meet non-functional requirements can result in systems that fail to satisfy user needs.

3.3 Hardware Requirements

Operating System supported by

1. Windows 7

2. Windows XP

3. Windows 8

Processor – Pentium IV or higher

RAM   -- 256 MB

Space on Hard Disk -- Minimum 512 MB

3.4 Software Requirements

 For developing the Application

1. Python

2. Django

3. MySQL

4. MySQL client

5. WampServer 2.4

Technologies and Languages used to Develop

-- Python

HOME PAGE:-

-XML

- JAVA

LOGIN PAGE:-

- XML

- JAVA

REGISTRATIONPAGE:-

- XML

- JAVA

BOOKINGS PAGE

- XML

- JAVA

REFERRALS PAGE

- XML

- JAVA

PROFILE PAGE

- XML

- JAVA

PREFERENCE PAGE

- XML

- JAVA

SP LOGIN PAGE

- XML

- JAVA

SP SIGN UP PAGE

- XML

- JAVA

SP BOOKING PAGE

- XML

- JAVA

CHAPTER 4

SYSTEM DESIGN

INTRODUCTION

This chapter provides the design phase of the Application. To design the project, we use the UML diagrams. The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.
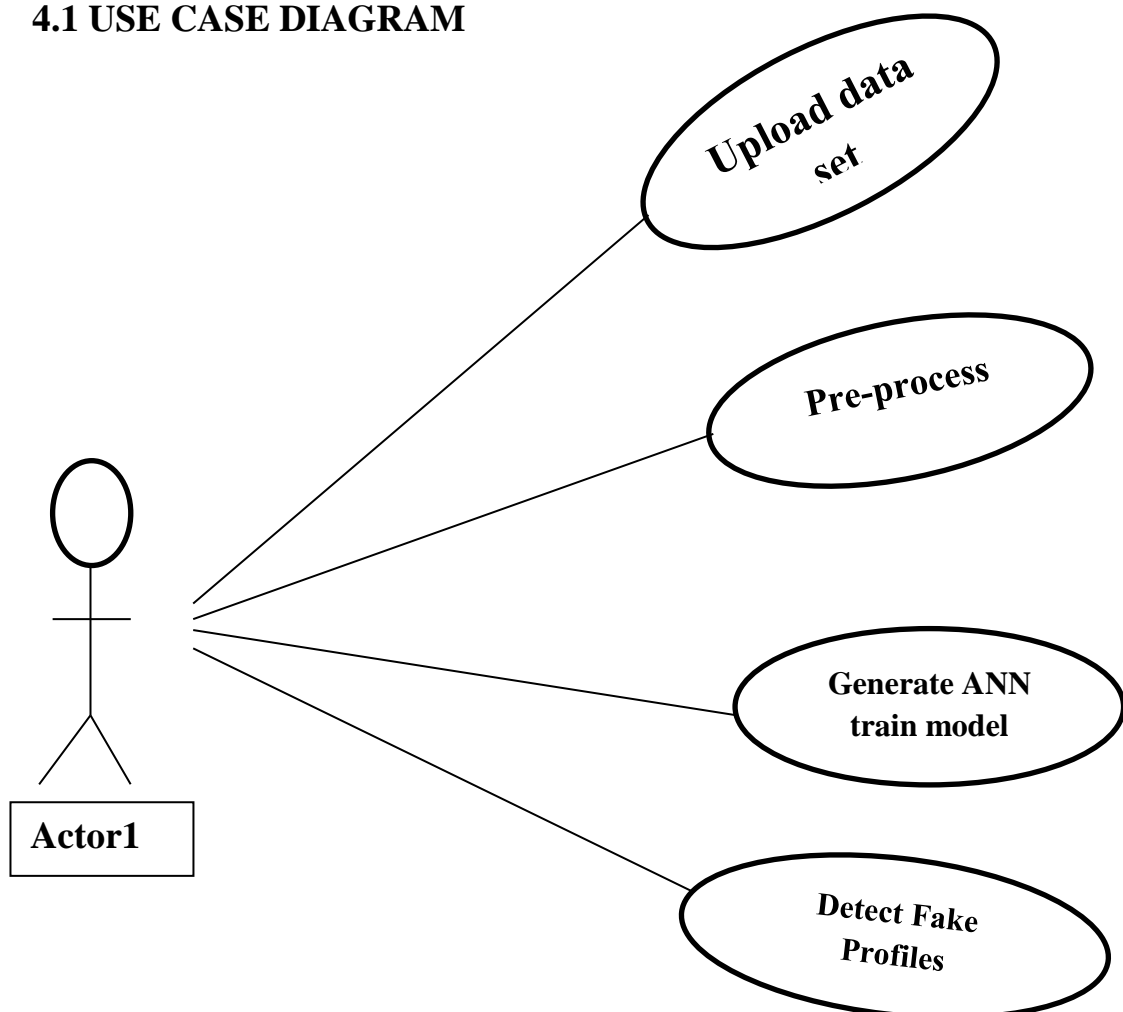
4.1 USE CASE DIAGRAM

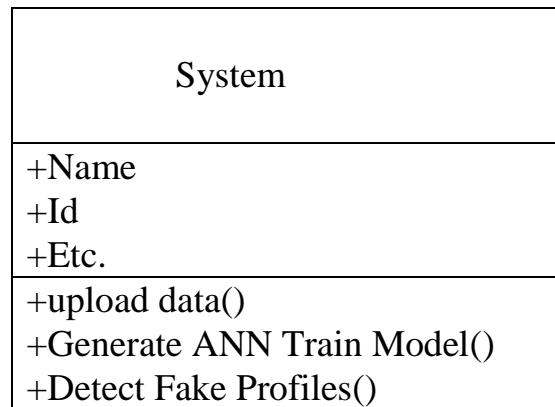

Fig 4.1 Use case Diagram

The use case diagram is used to represent all the functional use cases that are involved in the project.

The above diagram represents the main two **actors** in the project, they are

- User

4.2 CLASS DIAGRAM

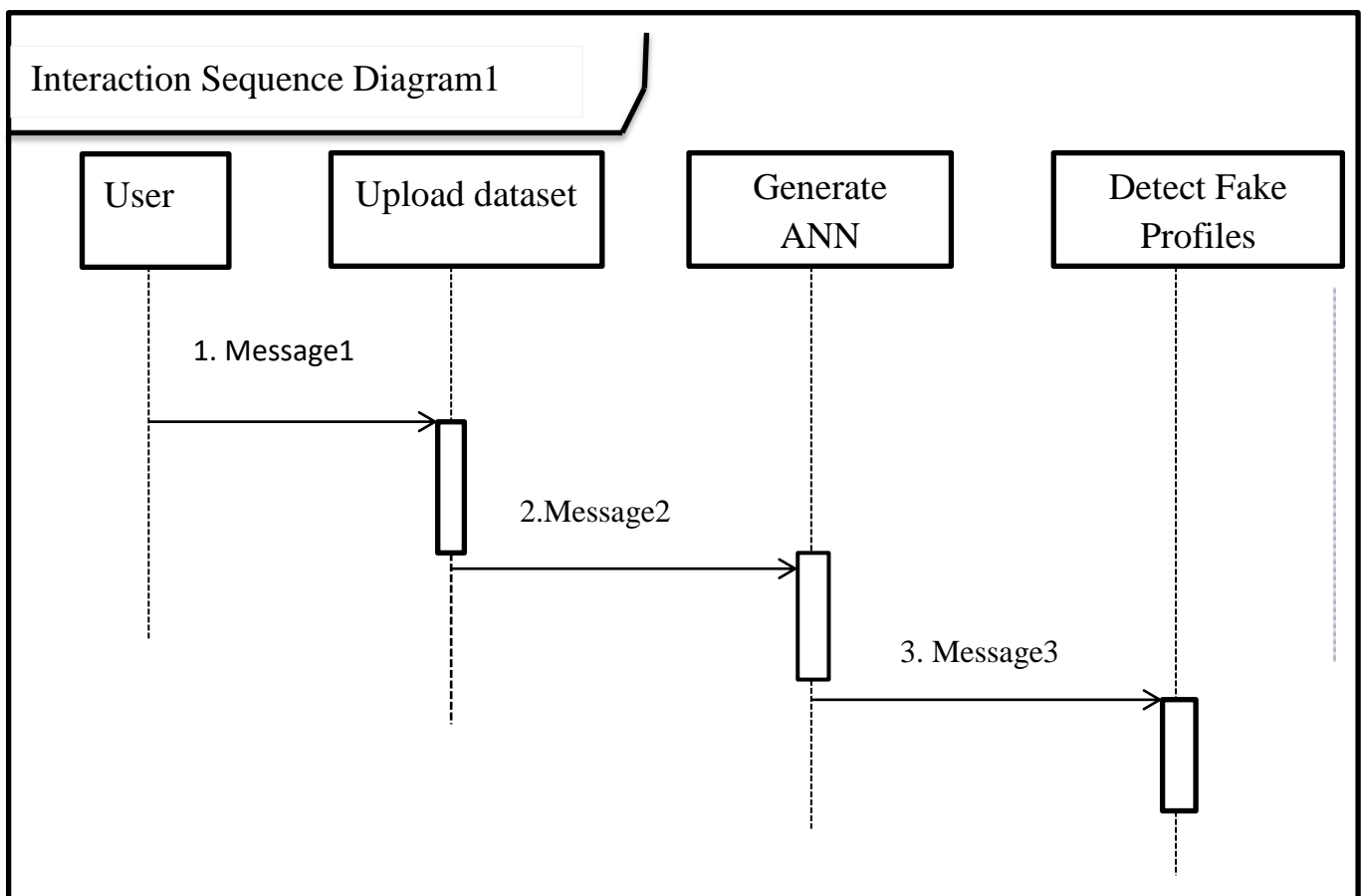| System |
| --- |
| +Name
+Id
+Etc. |
| +upload data()
+Generate ANN Train Model()
+Detect Fake Profiles() |

Fig3.2: class diagram

The above mentioned class diagram represents the Chatbot system workflow model. This diagram has class models with class names as

- User
- Admin
- Home screen

4.3 SEQUENCE DIAGRAM

Fig3.3: Sequence Diagram

4.4 ACTIVITY DIAGRAM
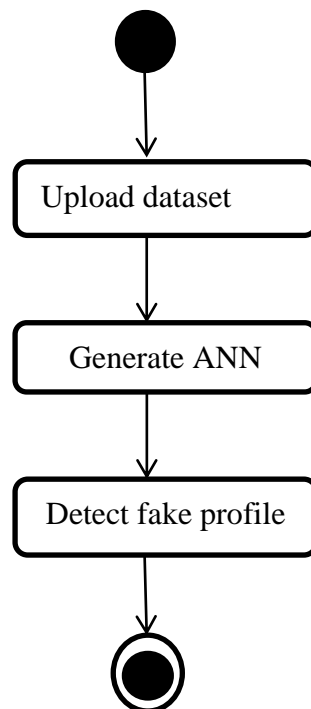
Fig3.4: Activity diagram

DATA DESIGN

Databases SQLite

Name
Use of Artificial

Table: SQLite Database

Tables

Name	Description
Users	Contains all the registered user details.
View Fake profiles deduction	All the registered service provider details.
Services	Contains all the types of services available.

Table: List of Database Table

CHAPTER 5

OVERVIEW OF PROJECT SOFTWARE

5.1 INTRODUCTION

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. Python is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library

5.2 PROJECT OVERVIEW

Python is a popular programming language. It was created by Guido van Possum, and released in 1991.

It is used for:

web development (server-side),

software development,

mathematics,

System scripting.

What can Python do

Python can be used on a server to create web applications.

Python can be used alongside software to create workflows.

Python can connect to database systems. It can also read and modify files.

Python can be used to handle big data and perform complex mathematics.

Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.).

Python has a simple syntax similar to the English language.

Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Python can be treated in a procedural way, an object-orientated way or a functional way.

Python Install

Many PCs and Macs will have python already installed.

To check if you have python installed on a Windows PC, search in the start bar for Python or run the following on the Command Line (cmd.exe):C:\Users\Your Name>python –version

To check if you have python installed on a Linux or Mac, then on Linux open the command line or on Mac open the Terminal and type:

Python --version

If you find that you do not have python installed on your computer, then you can download it for free from the following website: https://www.python.org/

Python Quick start

Python is an interpreted programming language, this means that as a developer you write Python (.py) files in a text editor and then put those files into the python interpreter to be executed.

The way to run a python file is like this on the command line:

C:\Users\Your Name>python helloworld.py

Where "helloworld.py" is the name of your python file.

Let's write our first Python file, called helloworld.py, which can be done in any text editor.

helloworld.py

Print ("Hello, World!")

Simple as that. Save your file. Open your command line, navigate to the directory where you saved your file, and run:

C:\Users\Your Name>python helloworld.py

The output should read:

Hello, World!

Congratulations, you have written and executed your first Python program.

The Python Command Line

To test a short amount of code in python sometimes it is quickest and easiest not to write the code in a file. This is made possible because Python can be run as a command line itself.

Type the following on the Windows, Mac or Linux command line:

C:\Users\Your Name>python

Or, if the "python" command did not work, you can try "py":

C:\Users\Your Name>py

From there you can write any python, including our hello world example from earlier in the tutorial:

C:\Users\Your Name>python

Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>print("Hello, World!")

Which will write "Hello, World!" in the command line:

C:\Users\Your Name>python

Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>print("Hello, World!")

Hello, World!

Whenever you are done in the python command line, you can simply type the following to quit the python command line interface:

exit()

Using the Python Interpreter

Invoking the Interpreter

The Python interpreter is usually installed as /usr/local/bin/python3.8 on those machines where it is available; putting /usr/local/bin in your UNIX shell's search path makes it possible to start it by typing the command:

python3.8

To the shell. 1 Since the choice of the directory where the interpreter lives is an installation option, other places are possible; check with your local Python guru or

system administrator. (E.g., /usr/local/python is a popular alternative location. On Windows machines where you have installed Python from the Microsoft Store, the python3.8 command will be available. If you have the py.exe launcher installed, you can use the py command. See Excursus: Setting environment variables for other ways to launch Python. Typing an end-of-file character (Control-D on UNIX, Control-Z on Windows) at the primary prompt causes the interpreter to exit with a zero exit status. If that doesn't work, you can exit the interpreter by typing the following command: quit ().

The interpreter's line-editing features include interactive editing, history substitution and code completion on systems that support the GNU Readline library. Perhaps the quickest check to see whether command line editing is supported is typing Control-P to the first Python prompt you get. If it beeps, you have command line editing; see Appendix Interactive Input Editing and History Substitution for an introduction to the keys. If nothing appears to happen, or if ^P is echoed, command line editing isn't available; you'll only be able to use backspace to remove characters from the current line.

The interpreter operates somewhat like the Unix shell: when called with standard input connected to a tty device, it reads and executes commands interactively; when called with a file name argument or with a file as standard input, it reads and executes a script from that file.

A second way of starting the interpreter is python -c command [arg] ..., which executes the statement(s) in command, analogous to the shells -c option. Since Python statements often contain spaces or other characters that are special to the shell, it is usually advised to quote command in its entirety with single quotes.

Some Python modules are also useful as scripts. These can be invoked using python -m module [arg] ..., which executes the source file for module as if you had spelled out its full name on the command line.

When a script file is used, it is sometimes useful to be able to run the script and enter interactive mode afterwards. This can be done by passing -i before the script.

All command line options are described in Command line and environment.

Argument Passing

When known to the interpreter, the script name and additional arguments thereafter are turned into a list of strings and assigned to the argv variable in the sys module. You can access this list by executing import sys. The length of the list is at least one; when no script and no arguments are given, sys.argv[0] is an empty string. When the script name is given as '-' (meaning standard input), sys.argv [0] is set to '-'. When -c command is used, sys.argv [0] is set to '-c'. When -m module is used, sys.argv [0] is set to the full name of the located module. Options found after -c command or -m module are not consumed by the Python interpreter's option processing but left in sys.argv for the command or module to handle.

Interactive Mode

When commands are read from a tty, the interpreter is said to be in interactive mode. In this mode it prompts for the next command with the primary prompt, usually three greater-than signs (>>>); for continuation lines it prompts with the secondary prompt, by default three dots (...). The interpreter prints a welcome message stating its version number and a copyright notice before printing the first prompt:

```
$ python3.8
```

Python 3.8 (default, Sep 16 2015, 09:25:04)

[GCC 4.8.2] on linux

Type "help", "copyright", "credits" or "license" for more information.

>>>

Continuation lines are needed when entering a multi-line construct. As an example, take a look at this if statement:

>>>

>>>the_world_is_flat = True

>>>ifthe_world_is_flat:

...   Print("Be careful not to fall off!")

Be careful not to fall off!

For more on interactive mode, see Interactive Mode.

The Interpreter and Its Environment

Source Code Encoding

By default, Python source files are treated as encoded in UTF-8. In that encoding, characters of most languages in the world can be used simultaneously in string literals, identifiers and comments — although the standard library only uses ASCII characters for identifiers, a convention that any portable code should follow. To display all these characters properly, your editor must recognize that the file is UTF-8, and it must use a font that supports all the characters in the file.

To declare an encoding other than the default one, a special comment line should be added as the first line of the file. The syntax is as follows:

-*- coding: encoding -*-

Where encoding is one of the valid codecs supported by Python.

For example, to declare that Windows-1252 encoding is to be used, the first line of your source code file should be:

-*- coding: cp1252 -*-

One exception to the first line rule is when the source code starts with a UNIX "shebang" line. In this case, the encoding declaration should be added as the second line of the file. For example:

#! /usr/bin/env python3

-*- coding: cp1252 -*-

5.3 Artificial Intelligence

"The science and engineering of making intelligent machines, especially intelligent computer programs". -John McCarthy-

Artificial Intelligence is an approach to make a computer, a robot, or a product to think how smart human think. AI is a study of how human brain think, learn, decide and work, when it tries to solve problems. And finally this study outputs intelligent software systems.The aim of AI is to improve computer functions which are related to human knowledge, for example, reasoning, learning, and problem-solving.

The intelligence is intangible. It is composed of

Reasoning

Learning

Problem Solving

Perception

Linguistic Intelligence

The objectives of AI research are reasoning, knowledge representation, planning, learning, natural language processing, realization, and ability to move and manipulate objects. There are long-term goals in the general intelligence sector.

Approaches include statistical methods, computational intelligence, and traditional coding AI. During the AI research related to search and mathematical optimization, artificial neural networks and methods based on statistics, probability, and economics, we use many tools. Computer science attracts AI in the field of science, mathematics, psychology, linguistics, philosophy and so on.

Applications of AI

· Gaming − AI plays important role for machine to think of large number of possible positions based on deep knowledge in strategic games. For example, chess,river crossing, N-queens problems and etc.

Natural Language Processing − Interact with the computer that understands natural language spoken by humans.

· Expert Systems − Machine or software provide explanation and advice to the users.

· Vision Systems − Systems understand, explain, and describe visual input on the computer.

· Speech Recognition − There are some AI based speech recognition systems have ability to hear and express as sentences and understand their meanings while a person talks to it. For example Siri and Google assistant.

 Handwriting Recognition − The handwriting recognition software reads the text written on paper and recognize the shapes of the letters and convert it into editable text.· Intelligent Robots − Robots are able to perform the instructions given by a human.

Major Goals

Knowledge reasoning

Planning

Machine Learning

 Natural Language Processing

 Computer Vision

 Robotics

5.4 Machine Learning

Introduction

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.

Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of

explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision-making processes based on data inputs.

Any technology user today has benefitted from machine learning. Facial recognition technology allows social media platforms to help users tag and share photos of friends. Optical character recognition (OCR) technology converts images of text into movable type. Recommendation engines, powered by machine learning, suggest what movies or television shows to watch next based on user preferences. Self-driving cars that rely on machine learning to navigate may soon be available to consumers.

Machine learning is a continuously developing field. Because of this, there are some considerations to keep in mind as you work with machine learning methodologies, or analyze the impact of machine learning processes.

In this tutorial, we'll look into the common machine learning methods of supervised and unsupervised learning, and common algorithmic approaches in machine learning, including the k-nearest neighbor algorithm, decision tree learning, and deep learning. We'll explore which programming languages are most used in machine learning, providing you with some of the positive and negative attributes of each. Additionally, we'll discuss biases that are perpetuated by machine learning algorithms, and consider what can be kept in mind to prevent these biases when building algorithms.

Machine Learning Methods

In machine learning, tasks are generally classified into broad categories. These categories are based on how learning is received or how feedback on the learning is given to the system developed.

Two of the most widely adopted machine learning methods are **supervised learning** which trains algorithms based on example input and output data that is labeled by humans, and **unsupervised learning** which provides the algorithm with no labeled data in order to allow it to find structure within its input data. Let's explore these methods in more detail.

Supervised Learning

In supervised learning, the computer is provided with example inputs that are labeled with their desired outputs. The purpose of this method is for the algorithm to be able to "learn" by comparing its actual output with the "taught" outputs to find errors, and modify the model accordingly. Supervised learning therefore uses patterns to predict label values on additional unlabeled data.

For example, with supervised learning, an algorithm may be fed data with images of sharks labeled as fish and images of oceans labeled as water. By being trained on this data, the supervised learning algorithm should be able to later identify unlabeled shark images as fish and unlabeled ocean images as water.

A common use case of supervised learning is to use historical data to predict statistically likely future events. It may use historical stock market information to anticipate upcoming fluctuations, or be employed to filter out spam emails. In supervised learning, tagged photos of dogs can be used as input data to classify untagged photos of dogs.

Unsupervised Learning

In unsupervised learning, data is unlabeled, so the learning algorithm is left to find commonalities among its input data. As unlabeled data are more abundant than labeled data, machine learning methods that facilitate unsupervised learning are particularly valuable.

The goal of unsupervised learning may be as straightforward as discovering hidden patterns within a dataset, but it may also have a goal of feature learning, which allows the computational machine to automatically discover the representations that are needed to classify raw data.

Unsupervised learning is commonly used for transactional data. You may have a large dataset of customers and their purchases, but as a human you will likely not be able to make sense of what similar attributes can be drawn from customer profiles and their types of purchases. With this data fed into an unsupervised learning algorithm, it may be determined that women of a certain age range who buy unscented soaps are likely to be pregnant, and therefore a marketing campaign related to pregnancy and baby

products can be targeted to this audience in order to increase their number of purchases.

Without being told a "correct" answer, unsupervised learning methods can look at complex data that is more expansive and seemingly unrelated in order to organize it in potentially meaningful ways. Unsupervised learning is often used for anomaly detection including for fraudulent credit card purchases, and recommender systems that recommend what products to buy next. In unsupervised learning, untagged photos of dogs can be used as input data for the algorithm to find likenesses and classify dog photos together.

Approaches

As a field, machine learning is closely related to computational statistics, so having background knowledge in statistics is useful for understanding and leveraging machine learning algorithms.

For those who may not have studied statistics, it can be helpful to first define correlation and regression, as they are commonly used techniques for investigating the relationship among quantitative variables. **Correlation** is a measure of association between two variables that are not designated as either dependent or independent. **Regression** at a basic level is used to examine the relationship between one dependent and one independent variable. Because regression statistics can be used to anticipate the dependent variable when the independent variable is known, regression enables prediction capabilities.

Approaches to machine learning are continuously being developed. For our purposes, we'll go through a few of the popular approaches that are being used in machine learning at the time of writing.

CHAPTER 6
SYSTEM LOWLEVEL DESIGN

This chapter mainly provides the overview on modules of the application, objectives of the project and a detailed project overview.

6.1 Modules of the Application:

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

Generate ANN Train Model: Admin will upload profile dataset to ANN algorithm to build train model. This train model can be used to predict fake or genuine account by taking new account test data.

View ANN Train Dataset: Using this module admin can view all dataset used to train ANN model.

User Module: Any user can use this application and enter test data of new account and call ANN algorithm. ANN algorithm will take new test data and applied train model to predict whether given test data contains fake or genuine details.

6.2 OBJECTIVES OF THE PROJECT

In this project, we outline the classes and libraries involved. We also discuss the sigmoid function and how are the weights determined and used. We also consider the parameters of the social network page which are the most important to our solution.

CHAPTER 7

INPUT AND OUTPUT DESIGN

7.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

What data should be given as input?

How the data should be arranged or coded?

The dialog to guide the operating personnel in providing input.

Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed

so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

7.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1.Designing computer output should proceed in an organized, well thought   out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

 3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

Convey information about past activities, current status or projections of the

Future.

Signal important events, opportunities, problems, or warnings.
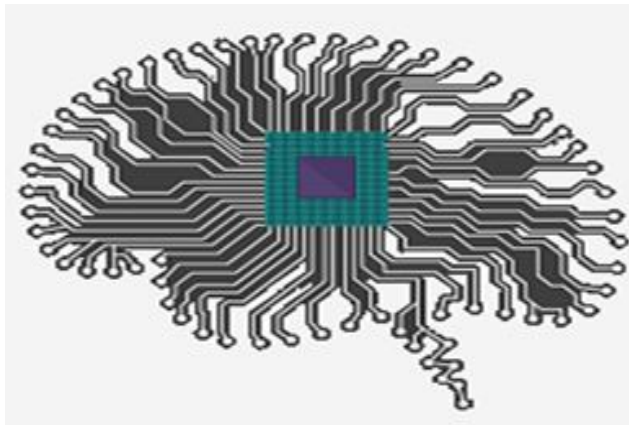
Trigger an action.

Confirm an action.

CHAPTER 8

IMPLEMENTATION

This chapter mainly provides the Algorithms and source code implementation.

8.1 ALGORITHMS

Artificial Neural Network

Artificial Neural Network Tutorial provides basic and advanced concepts of ANNs. Our

Artificial Neural Network Tutorial provides basic and advanced concepts of ANNs. Our Artificial Neural Network tutorial is developed for beginners as well as professions. The term "Artificial neural network" refers to a biologically inspired sub-field of artificial intelligence modeled after the brain. An Artificial neural network is usually a computational network based on biological neural networks that construct the structure of the human brain. Similar to a human brain has neurons interconnected to each other; artificial neural networks also have neurons that are linked to each other in various layers of the networks. These neurons are known as nodes.

Artificial neural network tutorial covers all the aspects related to the artificial neural network. In this tutorial, we will discuss ANNs, Adaptive resonance theory, Kohonen self-organizing map, Building blocks, unsupervised learning, Genetic algorithm, etc.
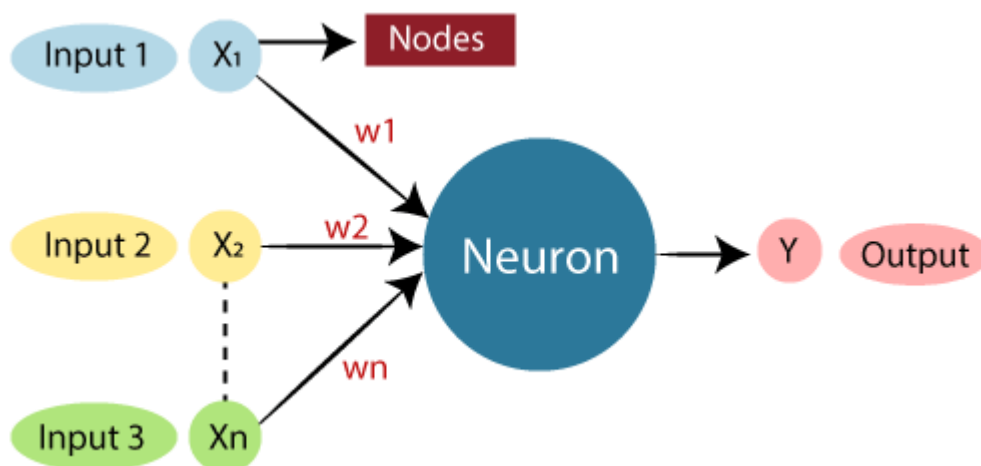
What is Artificial Neural Network?

The term "**Artificial Neural Network**" is derived from Biological neural networks that develop the structure of a human brain. Similar to the human brain that has neurons interconnected to one another; artificial neural networks also have neurons that are interconnected to one another in various layers of the networks. These neurons are known as nodes.
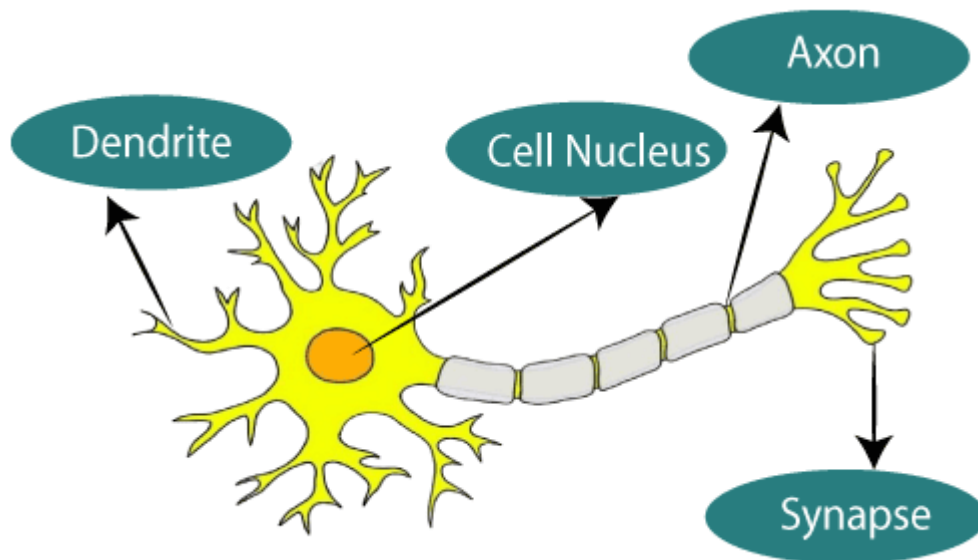
The given figure illustrates the typical diagram of Biological Neural Network.

The typical Artificial Neural Network looks something like the given figure.

Dendrites from Biological Neural Network represent inputs in Artificial Neural Networks, cell nucleus represents Nodes, synapse represents Weights, and Axon represents Output.

Relationship between Biological neural network and artificial neural network:

Biological Neural Network	Artificial Neural Network
Dendrites	Inputs
Cell nucleus	Nodes
Synapse	Weights
Axon	Output

An **Artificial Neural Network** in the field of **Artificial intelligence** where it attempts to mimic the network of neurons makes up a human brain so that computers will have an option to understand things and make decisions in a human-like manner. The artificial neural network is designed by programming computers to behave simply like interconnected brain cells. There are around 1000 billion neurons in the human brain. Each neuron has an association point somewhere in the range of 1,000 and 100,000. In the human brain, data is stored in such a manner as to be distributed, and we can extract more than one piece of this data when necessary from our memory parallels. We can say that the human brain is made up of incredibly amazing parallel processors. We can understand the artificial neural network with an example, consider an example of a digital logic gate that takes an input and gives an output. "OR" gate, which takes two inputs. If one or both the inputs are "On," then we get "On" in output. If both the inputs are "Off," then we get "Off" in output. Here the output depends upon input. Our brain does not perform the same task. The outputs to inputs relationship keep changing because of the neurons in our brain, which are "learning.

The architecture of an artificial neural network:

To understand the concept of the architecture of an artificial neural network, we have to understand what a neural network consists of. In order to define a neural network that consists of a large number of artificial neurons, which are termed units arranged in a sequence of layers. Let's us look at various types of layers available in an artificial neural network.

Artificial Neural Network primarily consists of three layers:
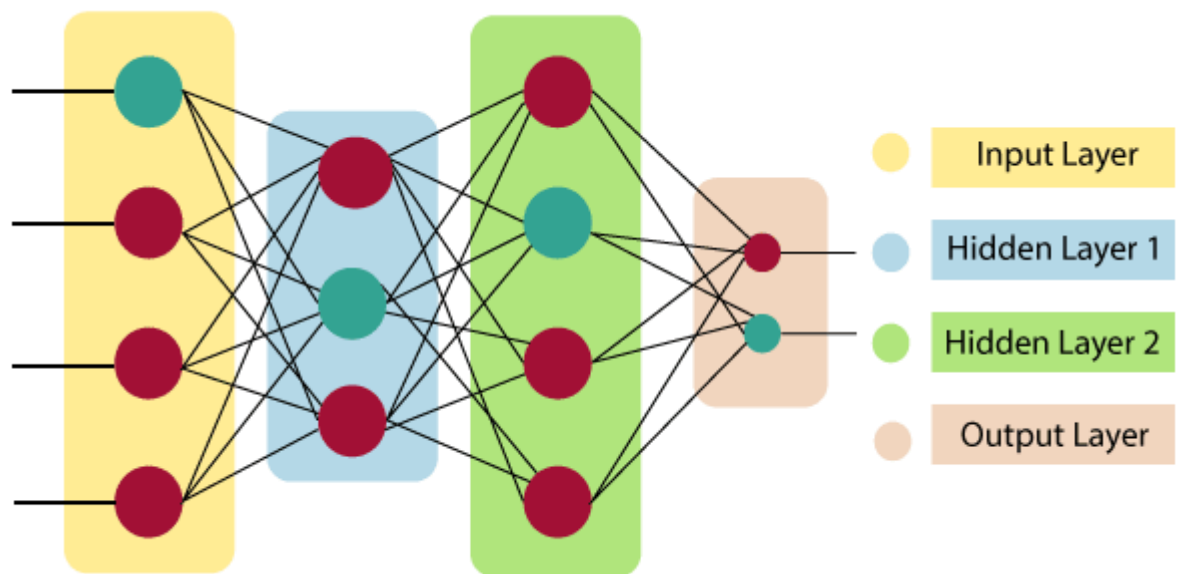
Fig7.1: Architecture of ANN

Input Layer:

As the name suggests, it accepts inputs in several different formats provided by the programmer.

Hidden Layer:

The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.

Output Layer:

The input goes through a series of transformations using the hidden layer, which finally results in output that is conveyed using this layer.

The artificial neural network takes input and computes the weighted sum of the inputs and includes a bias. This computation is represented in the form of a transfer function.

$$\sum_{i=1}^{n} Wi * Xi + b$$

It determines weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer. There are distinctive activation functions available that can be applied upon the sort of task we are performing.

Advantages of Artificial Neural Network (ANN)

Parallel processing capability:

Artificial neural networks have a numerical value that can perform more than one task simultaneously.

Storing data on the entire network:

Data that is used in traditional programming is stored on the whole network, not on a database. The disappearance of a couple of pieces of data in one place doesn't prevent the network from working.

Capability to work with incomplete knowledge:

After ANN training, the information may produce output even with inadequate data. The loss of performance here relies upon the significance of missing data.

Having a memory distribution:

For ANN is to be able to adapt, it is important to determine the examples and to encourage the network according to the desired output by demonstrating these examples to the network. The succession of the network is directly proportional to the

chosen instances, and if the event can't appear to the network in all its aspects, it can produce false output.

Having fault tolerance:

Extortion of one or more cells of ANN does not prohibit it from generating output, and this feature makes the network fault-tolerance.

Disadvantages of Artificial Neural Network:

Assurance of proper network structure:

There is no particular guideline for determining the structure of artificial neural networks. The appropriate network structure is accomplished through experience, trial, and error.

Unrecognized behaviour of the network:

It is the most significant issue of ANN. When ANN produces a testing solution, it does not provide insight concerning why and how. It decreases trust in the network.

Hardware dependence:

Artificial neural networks need processors with parallel processing power, as per their structure. Therefore, the realization of the equipment is dependent.

Difficulty of showing the issue to the network:

ANNs can work with numerical data. Problems must be converted into **The duration of the network is unknown:**

Numerical values before being introduced to ANN. The presentation mechanism to be resolved here will directly impact the performance of the network. It relies on the user's abilities

The network is reduced to a specific value of the error, and this value does not give us optimum results. Science artificial neural networks that have stepped into the world in the mid-20th century are exponentially developing. In the present time, we have investigated the pros of artificial neural networks and the issues encountered in the course of their utilization. It should not be overlooked that the cons of ANN networks,

which are a flourishing science branch, are eliminated individually, and their pros are increasing day by day. It means that artificial neural networks will turn into an irreplaceable part of our lives progressively important.

How do artificial neural networks work?

Artificial Neural Network can be best represented as a weighted directed graph, where the artificial neurons form the nodes. The association between the neurons outputs and neuron inputs can be viewed as the directed edges with weights. The Artificial Neural Network receives the input signal from the external source in the form of a pattern and image in the form of a vector. These inputs are then mathematically assigned by the notations x(n) for every n number of inputs.
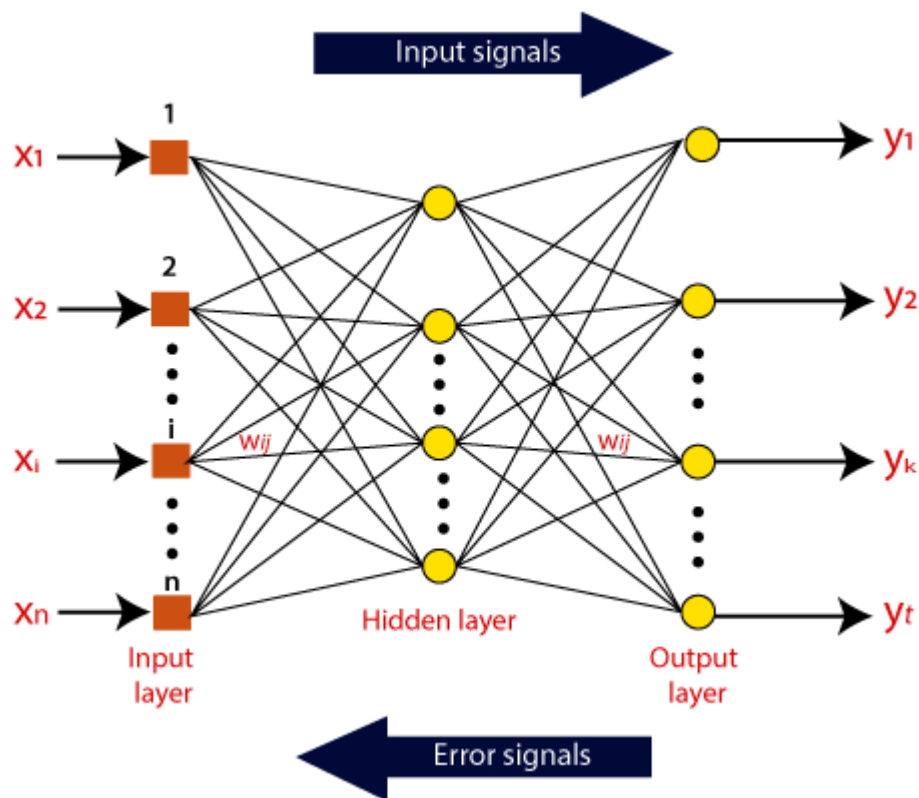
Afterward, each of the input is multiplied by its corresponding weights (these weights are the details utilized by the artificial neural networks to solve a specific problem). In general terms, these weights normally represent the strength of the interconnection between neurons inside the artificial neural network. All the weighted inputs are summarized inside the computing unit.

If the weighted sum is equal to zero, then bias is added to make the output non-zero or something else to scale up to the system's response. Bias has the same input and weight equals to 1. Here the total of weighted inputs can be in the range of 0 to positive infinity. Here, to keep the response in the limits of the desired value, a certain maximum value is benchmarked, and the total of weighted inputs is passed through the activation function.

The activation function refers to the set of transfer functions used to achieve the desired output. There is a different kind of the activation function, but primarily either linear or non-linear sets of functions. Some of the commonly used sets of activation functions are the Binary, linear, and Tan hyperbolic sigmoidal activation functions. Let us take a look at each of them in details:

Binary:

In binary activation function, the output is either a one or a 0. Here, to accomplish this, there is a threshold value set up. If the net weighted input of neurons is more than 1, then the final output of the activation function is returned as one or else the output is returned as 0.

Sigmoidal Hyperbolic:

The Sigmoidal Hyperbola function is generally seen as an "**S**" shaped curve. Here the tan hyperbolic function is used to approximate output from the actual net input. The function is defined as:

F(x) = (1/1 + exp (-????x))

Where???? is considered the Steepness parameter.

Types of Artificial Neural Network:

There are various types of Artificial Neural Networks (ANN) depending upon the human brain neuron and network functions, an artificial neural network similarly perform tasks. The majority of the artificial neural networks will have some similarities with a more complex biological partner and are very effective at their expected tasks. For example, segmentation or classification.

Feedback ANN:

In this type of ANN, the output returns into the network to accomplish the best-evolved results internally. As per the **University of Massachusetts**, Lowell Centre for Atmospheric Research. The feedback networks feed information back into itself and are well suited to solve optimization issues. The Internal system error corrections utilize feedback ANNs.

Feed-Forward ANN:

A feed-forward network is a basic neural network comprising of an input layer, an output layer, and at least one layer of a neuron. Through assessment of its output by reviewing its input, the intensity of the network can be noticed based on group behavior of the associated neurons, and the output is decided. The primary advantage of this network is that it figures out how to evaluate and recognize input patterns

8.2 SOURCE CODE

This chapter mainly provides the source code and implementation of the project.

from django.shortcuts import render

from django.template import RequestContext

from django.contrib import messages

from django.http import HttpResponse

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

```python
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':

            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:/FakeProfile/Profile/dataset/dataset.txt')

    balance_data = balance_data.abs()
```

```python
    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('t1', False)

      input                                                                   =
'Account_Age,Gender,User_Age,Link_Desc,Status_Count,Friend_Count,Location,Lo
cation_IP\n';

      input+=data+"\n"

      f = open("C:/FakeProfile/Profile/dataset/test.txt", "w")

      f.write(input)

      f.close()

      test = pd.read_csv('C:/FakeProfile/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('CNN Neural Network Model Summary: ')

    print(model.summary())

    model.fit(train_x, train_y, verbose=2, batch_size=5, epochs=200)
```

```python
    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:/FakeProfile/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)
```
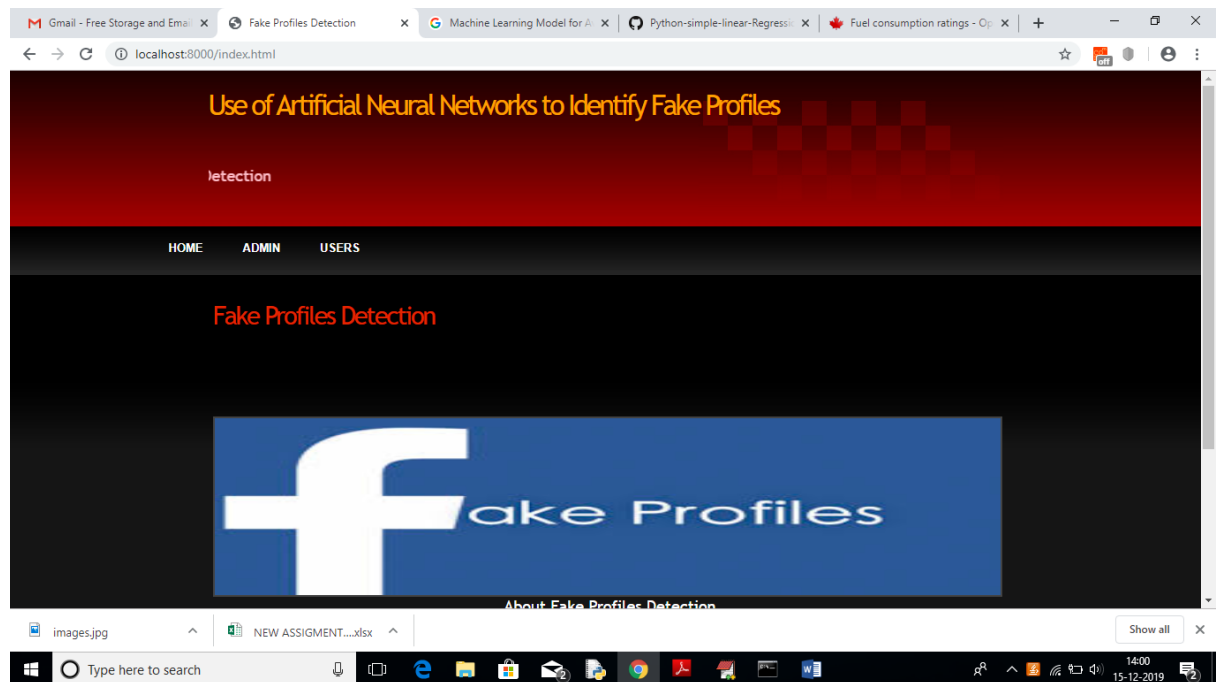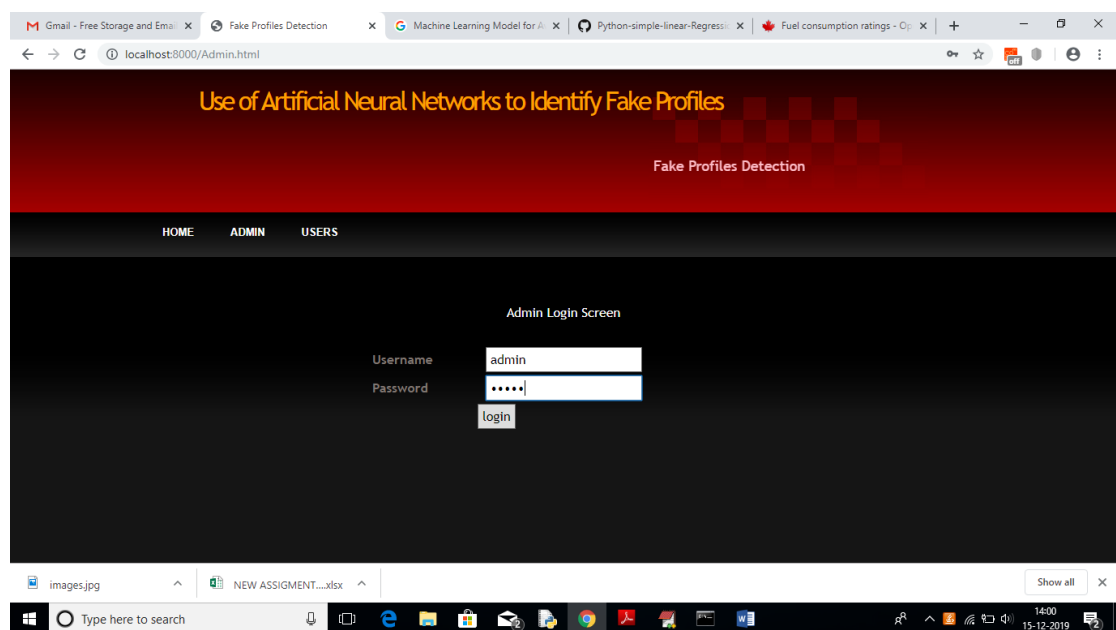
8.3 Output Screens

8.3.1User Login Screen:

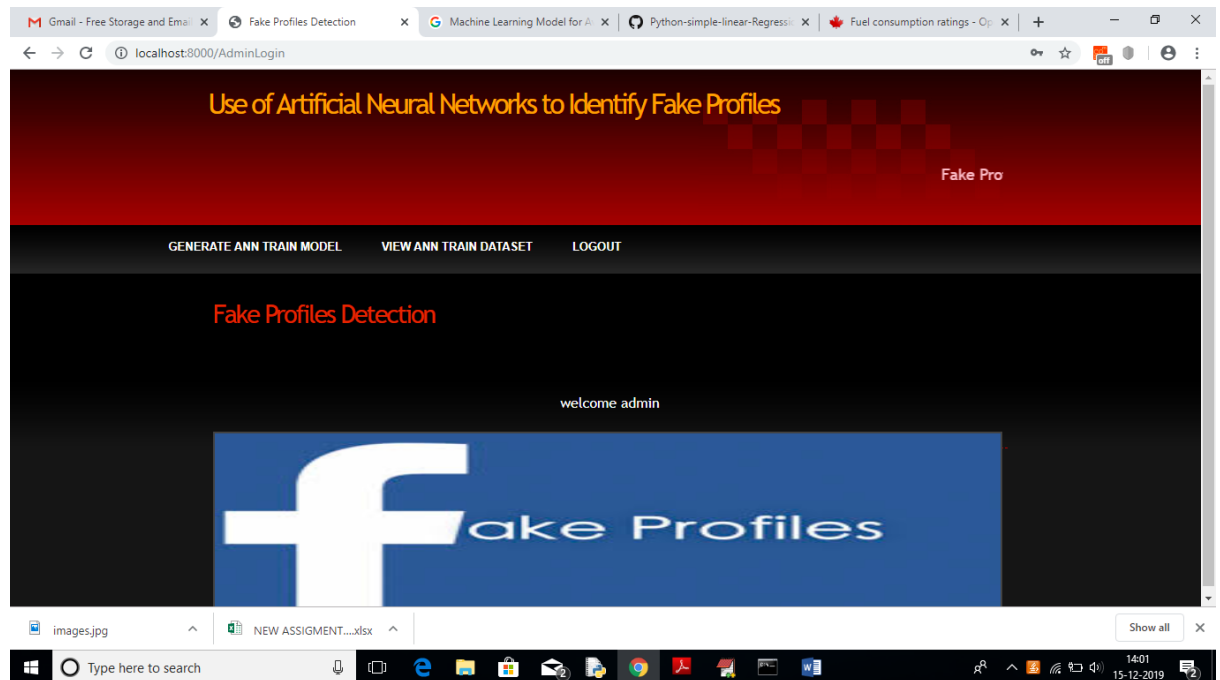Deploy this application on DJANGO server and then run in browser enter URL

as 'http://localhost:8000/index.html' to get below screen

In above screen click on 'ADMIN' link to get below login screen

In above screen enter admin and admin as username and password to login as admin. After login will get below screen
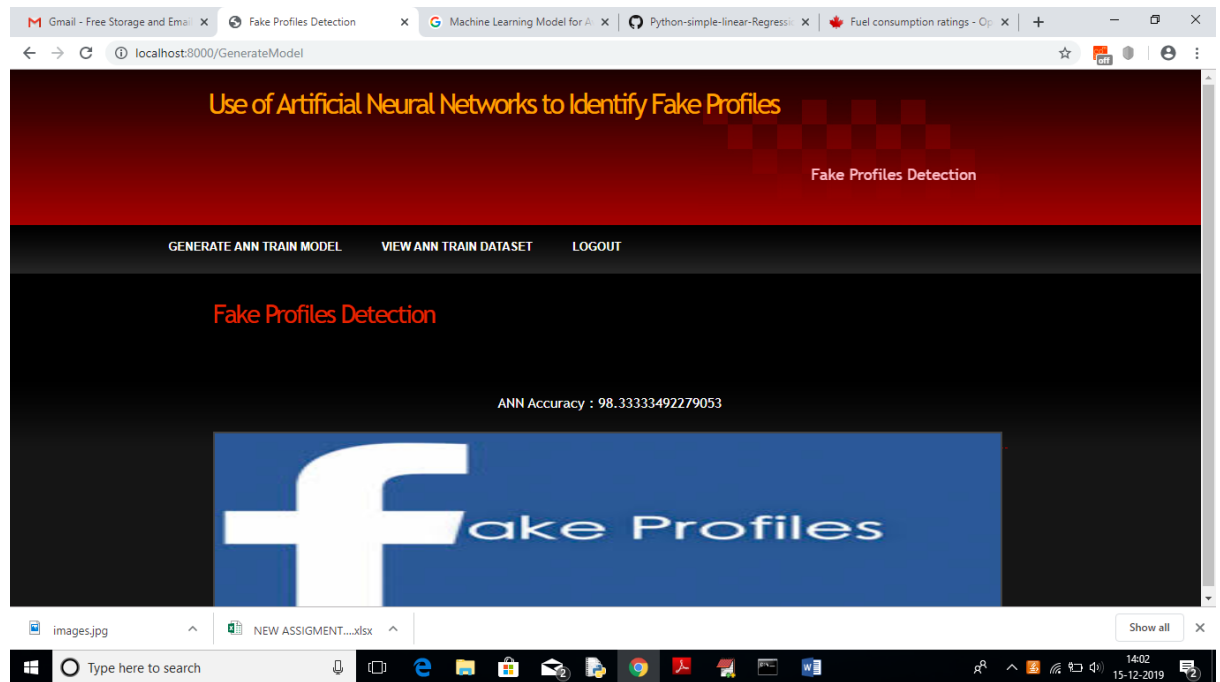

In above screen click on 'Generate ANN Train Model' to generate training model on dataset. After clicking on that link you can see server console to check ANN processing details with accuracy

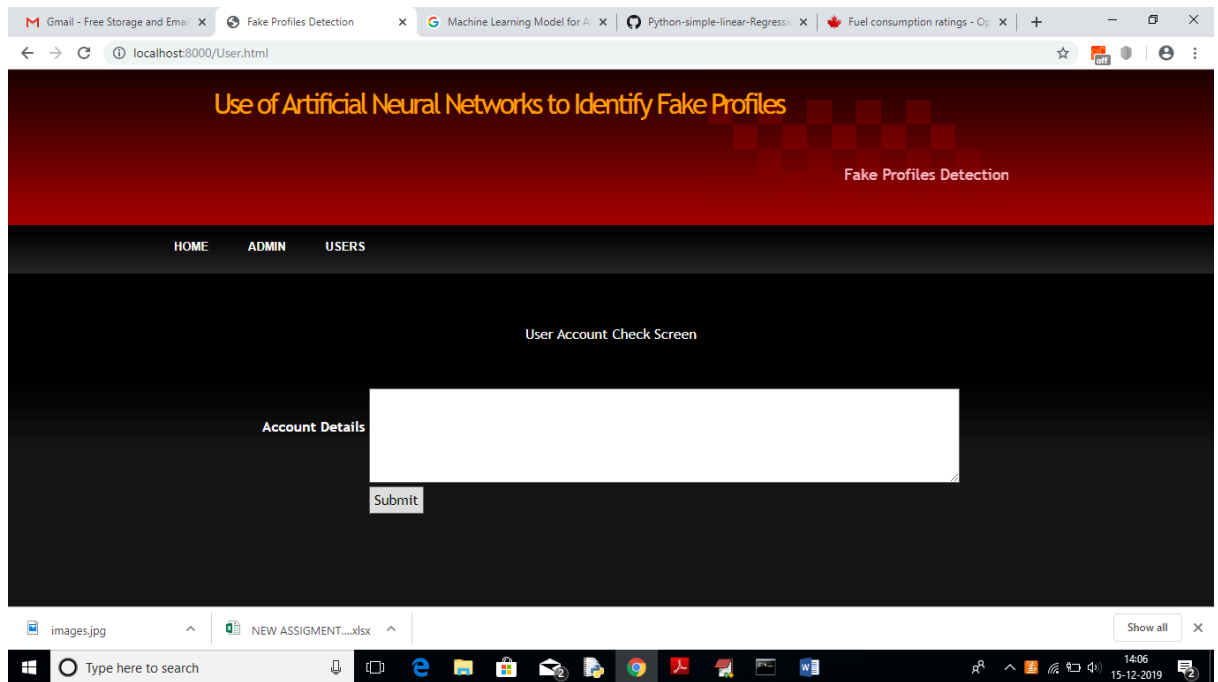In above black console we can see all ANN details.

In above screen we can see ANN got 98% accuracy to train all Facebook profile. Now click on 'View Ann Train Dataset' link to view all dataset details

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

In above screen we can see all train data and scroll down to view all records. Now ANN train model is ready and you can logout and click on 'User' link to get below screen.

In above screen enter some test account details to get prediction/identification from ANN. You can use below records to check
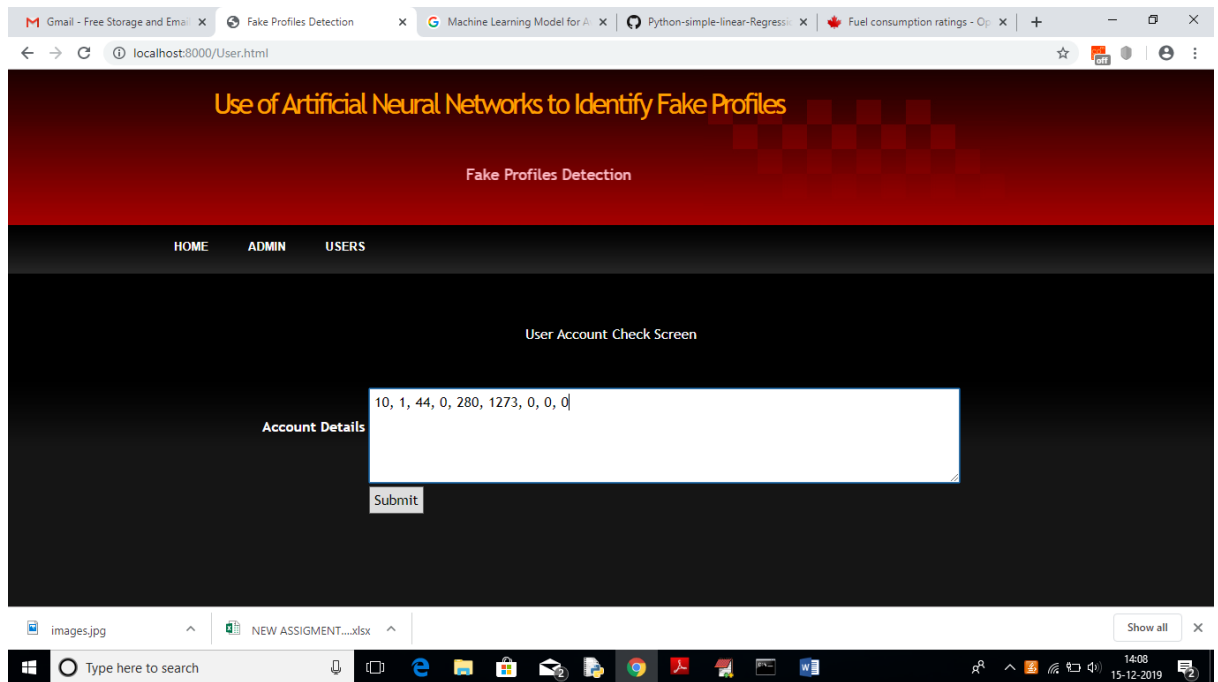
10, 1, 44, 0, 280, 1273, 0, 0
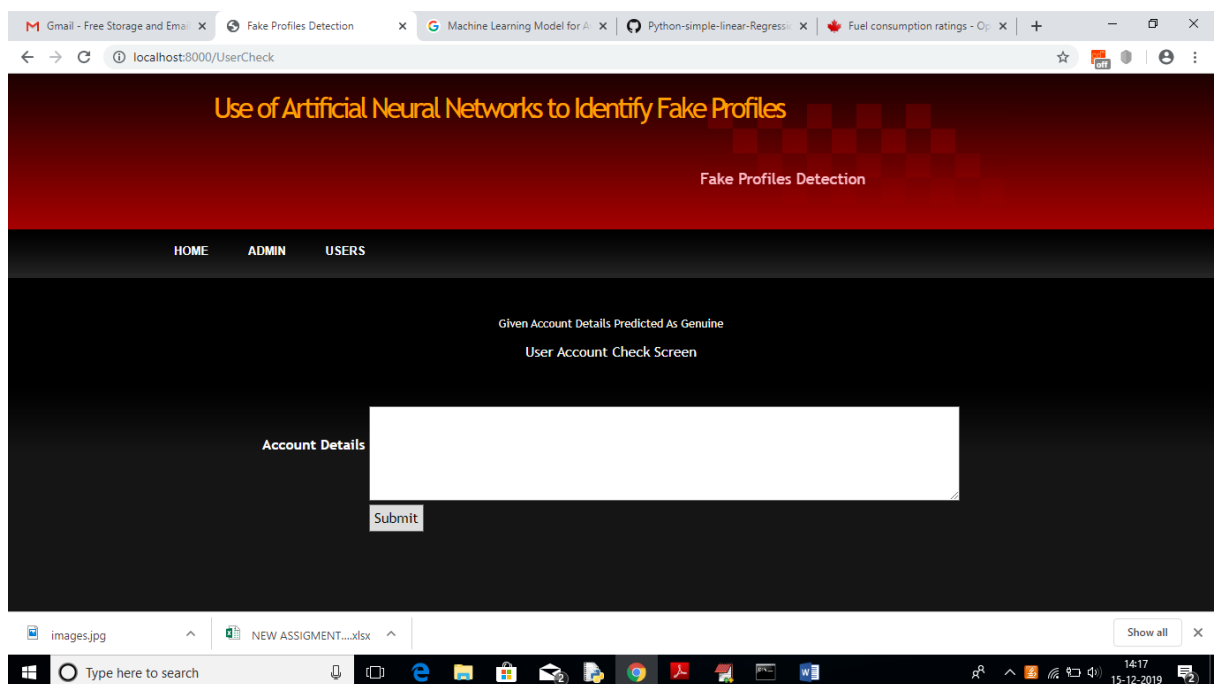
10, 0, 54, 0, 5237, 241, 0, 0

7, 0, 42, 1, 57, 631, 1, 1

7, 1, 56, 1, 66, 623, 1, 1

For above input will get below result

In above screen we can see the result predicted as genuine account

For above account details we got below result

In above screen we got result as fake for given account data.

CHAPTER 9

TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

9.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

9.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at   exposing the problems that arise from the combination of components.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

9.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input   : identified classes of valid input must be accepted.

Invalid Input   : identified classes of invalid input must be rejected.

Functions   : identified functions must be exercised.

Output   : identified classes of application outputs must be exercised.

Systems/Procedures   : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

9.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

9.5 Software Testing

Software testing is the process of validating and verifying that a software application meets the technical requirements which are involved in its design and development. It is also used to uncover any defects/bugs that exist in the application. It assures the

quality of the software. There are many types of testing software viz., manual testing, unit testing, black box testing, performance testing, stress testing, regression testing, white box testing etc. Among these performance testing and load testing are the most important one for an android application and next sections deal with some of these types.

9.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

9.7 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

9.8 Performance Testing

Performance testing is executed to determine how fast a system or sub-system performs under a particular workload. It can also serve to validate and verify other quality attributes of the system such as scalability, reliability and resource usage.

9.9 Load Testing

Load testing is primarily concerned with testing that can continue to operate under specific load, whether that is large quantities of data or a large number of users.

9.10 Manual Testing

Manual Testing is the process of manually testing software for defects. Functionality of this application is manually tested to ensure the correctness. Few examples of test case for Manual Testing are discussed later in this chapter.

Test Case 1	
Test Case Name	Empty login fields testing
Description	In the login screen if the username and password fields are empty
Output	Login fails showing an alert box asking to enter username and password.

Table 6:1 Test Case for Empty Login Field

Test Case 2	
Test Case Name	Wrong login fields testing
Description	A unique username and password are set by administrator. On entering wrong username or password gives.
Output	Login fails showing an alert box username or password incorrect.

Table 6:2 Test Case for Wrong Login Fields

Test Case 3	
Test Case Name	User Signup Fails.
Description	User signup need to provide all data.
Output	Signup Fails and an alert message appears asking to enter valid email and name.

Table6:3 Test Case for Signup fail

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

All field entries must work properly.

Pages must be activated from the identified link.

The entry screen, messages and responses must not be delayed.

Features to be tested

Verify that the entries are of the correct format

No duplicate entries should be allowed

All links should take the user to the correct page.

9.11 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 10

RESULTS AND CHALLENGES

10.1 Results

The current android application is developed using Xml, Java, SQL with Firebase connectivity. It can be used by every individual who are in a need of fulfilling their household services.

At the time of submission of my application was capable of doing the following:

- Displaying the home screen with different fragments.

- Authentication of user by using login screen using Firebase.

- Home screen to display based on user or service provider.

- After successful login of user, they can choose the service and book a slot of their particular service provider from the displayed list.

- Add, update, view, and delete the user details.

- After successful login of service provider, they can view all the bookings that are booked by the users and can attend them one by one.

- Service provider can also set his preferences to not available, if he's too busy or many users had already booked him.

- Service provider has the ability to change their particular radius of location for servicing.

- He can set up to 10 km radius.

- Logout and end the session.

10.2 Challenges

- Understanding the connections of SQLite Database is a tricky part and confusing when dealing with multiple tables within a database.

- Making exact orientation API design levels was a difficult task as there are many types of devices like desktop, tablet, mobile with varying screen size and resolutions.

- Implementing synchronization with Firebase was a challenging task.

- Learning different technologies and frameworks with little guidance.

CHAPTER 11
CONCLUSION

11.1   Conclusion

we use machine learning, namely an artificial neural network to determine what are the chances that a friend request is authentic are or not. Each equation at each neuron (node) is put through a Sigmoid function. We use a training data set by Facebook or other social networks. This would allow the presented deep learning algorithm to learn the patterns of bot behavior by back propagation, minimizing the final cost function and adjusting each neuron's weight and bias.

11.2 Scope for future work

- Each input neuron would be a different, previously chosen feature of each profile converted into a numerical value (e.g., gender as a binary number, female 0 and male 1) and if needed, divided by an arbitrary number (e.g., age is always divided by 100) to minimize one feature having more influence on the result than the other. The neurons represent nodes. Each node would be responsible for exactly one decision-making process

BIBLIOGRAPHY

Code snippets for any errors
http://stackoverflow.com/

Android Development Guide
https://www.udemy.com/android

Xml and Layout Guide
https://ww w.androidhive.com/

Connecting to Firebase Docs
https://firebase.google.com

Software Testing
http://en.wikipedia.org/wiki/Software_testing

Manual Testing
http://en.wikipedia.org/wiki/Manual_testing

Performance Testing
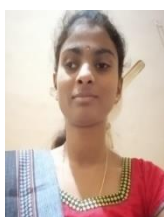http://en.wikipedia.org/wiki/Software_performance_testing

BATCH MEMBERS

NAME   : N.UDAYASRI

DOB   : 16-08-2001

E-MAIL   :narapaneniudayasri123@gmail.com

PHONE NO: +91 9959392620

NAME   : V.PRAVEENA

DOB   : 28-06-2002

E-MAIL   :praveenareddyveeram@gmail.com

PHONE NO: +919346484901

NAME   : P.SRAVANI

DOB   : 22-06-2002

E-MAIL   :pachasravani2219@gmail.com

PHONE NO: +91 6305910907

NAME   : M.SUPRIYA

DOB   : 22-06-2002

E-MAIL   :supriyamukkara789@gmail.com

PHONE NO: +91 7981080670

NAME   : K.SAI SINDURA

DOB   : 01-08-1995

E-MAIL   : Sindhukatari09@gmail.com

PHONE NO: +91 9581643196

PROJECT GUIDE

NAME   : D.ESWARAIAH

DESIGNATION :ASSISTANT PROFESSOR

E-MAIL   : d.eswaraiah515@gmail.com

PHONE NO   :+91 88860 14512